INVESTIGATING INERTIAL MEASUREMENT FOR HUMAN-SCALE MOTION TRACKING

by

Eric Allen Johnson

A dissertation submitted to the faculty of
The University of Utah
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

Department of Mechanical Engineering

The University of Utah

August 2011

The University of Utah Graduate School

STATEMENT OF DISSERTATION APPROVAL

The dissertation of **Eric Allen Johnson**

has been approved by the following supervisory committee members:

Stacy J. Morris Bamberg	, Chair	**5/7/2011** Date Approved
Mark A. Minor	, Member	**6/13/2011** Date Approved
John M. Hollerbach	, Member	**6/13/2011** Date Approved
Eberhard Bamberg	, Member	**5/8/2011** Date Approved
William Provancher	, Member	**6/13/2011** Date Approved

and by **Timothy Ameel**, Chair of the Department of **Mechanical Engineering**

and by Charles A. Wight, Dean of The Graduate School.

ABSTRACT

The need for position and orientation information in a wide variety of applications has led to the development of equally varied methods for providing it. Amongst the alternatives, inertial navigation is a solution that offers self-contained operation and provides angular rate, orientation, acceleration, velocity, and position information. Until recently, the size, cost, and weight of inertial sensors has limited their use to vehicles with relatively large payload capacities and instrumentation budgets. However, the development of microelectromechanical system (MEMS) inertial sensors now offers the possibility of using inertial measurement in smaller, even human-scale, applications.

Though much progress has been made toward this goal, there are still many obstacles. While operating independently from any outside reference, inertial measurement suffers from unbounded errors that grow at rates up to cubic in time. Since the reduced size and cost of these new miniaturized sensors comes at the expense of accuracy and stability, the problem of error accumulation becomes more acute. Nevertheless, researchers have demonstrated that useful results can be obtained in real-world applications.

The research presented herein provides several contributions to the development of human-scale inertial navigation. A calibration technique allowing complex sensor models to be identified using inexpensive hardware and linear solution techniques has been developed. This is shown to provide significant improvements in the accuracy of the calibrated outputs from MEMS inertial sensors. Error correction algorithms based on easily identifiable characteristics of the sensor outputs have also been developed. These are demonstrated in both one- and three-dimensional navigation. The results show significant improvements in the levels of accuracy that can be obtained using these inexpensive sensors. The al-

gorithms also eliminate empirical, application-specific simplifications and heuristics, upon which many existing techniques have depended, and make inertial navigation a more viable solution for tracking the motion around us.

To my wife and family, my support and inspiration.

CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 Project motivation

1.1.1 Overview of inertial measurement

Inertial measurement uses the output of accelerometers and angular rate sensors, or gyros, to provide information about the motion of a body. Full spatial motion can be estimated using three orthogonal accelerometer and gyro sensor axes. To do this, the sensor package is first calibrated and positioned on the body of interest. The initial position and orientation of the body are then provided relative to a reference frame. As the body moves, the angular rates output by the gyros are integrated once to provide an estimate of the body's orientation. This orientation estimate is used to resolve the acceleration measured by the accelerometers into the reference frame. The influence of gravity can then be removed from the accelerometer outputs and the remaining acceleration vector is integrated twice to provide estimates of the body's velocity and position in the reference frame. This process of converting the angular rate and acceleration readings from the sensors to orientation, position, and velocity estimates is referred to as inertial navigation.

At each stage of this process, errors degrade the accuracy of the results and grow without bound in the absence of additional references. However, as discussed in Chapter 2, prior work in the field has shown that given sufficiently accurate sensors, careful calibration, and sophisticated processing methods, inertial navigation can provide useful levels of accuracy. It offers a motion tracking solution that is totally self-contained and nonjammable. Since it does not depend on outside references, like global positioning

systems, it also does not suffer from signal blockage and can be used in any location. Furthermore, inertial navigation provides position, velocity, acceleration, orientation, and angular rate information all from a single source with update rates and resolution limited only by available processing power.

These advantages have motivated research regarding the use of inertial navigation for nearly a century. As early as the 1920s, inertial navigation was used in aviation and it was later adapted and applied to ballistics and space exploration as those fields developed [1]. Throughout this period, valuable knowledge and sophisticated techniques for inertial navigation at the global and even celestial scale have been developed [1–5]. More recently, as inertial sensors based on microelectromechanical systems (MEMS) have steadily improved [6–9], their reduced cost, size, and weight has motivated the exploration of inertial measurement in applications such as unmanned aerial [10] and underwater vehicles [11, 12], mobile robots [13–15], and human motion tracking for a variety of applications [16–27].

1.1.2 Specific motivations for this project

The research presented in this document was motivated by potential applications of inertial measurement in human motion tracking, mobile robotics, slow-moving micro-air vehicles, and other indoor motion-tracking applications on similar scales. Many of these applications share similar dynamics and present difficulties in defining system models that accurately account for their motion. MEMS-based inertial measurement units offer the potential to provide a motion tracking solution that would offer significant improvements in versatility and cost relative to existing motion tracking methods used in these applications.

There are still significant obstacles in the development of this type of inertial navigation solution. MEMS-based inertial sensors offer significantly less accuracy than traditional navigation-grade sensors. Their signal-to-noise ratios are relatively poor, nonlinearities and anisotropic effects are more prominent, and the gyros have higher sensitivity to acceleration [1, Ch. 7]. Parameter stability over time and given temperature variations is

also problematic, requiring frequent calibration and in-use updates. Left unchecked, the combination of these factors leads to unbounded error growth that is roughly linear in time for the orientation estimate and up to cubic in time for the position estimate. This rate of error growth can render the accuracy of the inertial measurement unit's output entirely useless in a matter of seconds. Until MEMS inertial sensors with improved accuracy are developed, specialized calibration and data processing methods capable of reducing the error growth rates will be necessary to obtain useful navigation information.

1.2 Research objectives

The objectives of this research project were to develop calibration techniques, data processing methods, and the hardware required to investigate the feasibility of using MEMS inertial sensors for human-scale motion tracking. Specific steps to accomplish these objectives included:

- Development of appropriate sensor models and calibration methods to characterize and compensate for the nonideal outputs from MEMS inertial sensors
- Design of sensor-based, sensor-level algorithms that would account for their accuracy limitations and reduce error growth rates
- Integration of the sensor-level error correction algorithms into complete inertial navigation solutions
- Validation of the effectiveness of the resulting navigation algorithms

It was hypothesized that by utilizing appropriate sensor models, calibration techniques, and error correction algorithms, useful results for human-scale motion tracking could be obtained.

1.3 Key contributions

1.3.1 Hardware development

Testing and development of the algorithms used for inertial navigation required both an inertial measurement unit and the hardware necessary to calibrate and demonstrate its

use in applications of interest. During preliminary tests, the ungainly conglomeration of manufacturer's evaluation boards shown in Figure 1.1 was used with the small and limited mobile robot seen in Figure 1.2.

One of the contributions from this project was the organization and supervision of an undergraduate research team that designed and assembled the circuit boards in the custom MEMS inertial measurement unit shown in Figure 1.3. The reduced size and weight of this inertial measurement unit allow motion tracking experiments to be conducted in a much wider range of applications. The specifications for this inertial measurement unit are provided in Chapter 4.

Second, a mobile robot platform was designed and constructed specifically for inertial measurement research to enable more advanced experiments to be conducted. Renderings of the solid model are shown in Figure 1.4 and the constructed robot is shown prepared to conduct a three-dimensional motion tracking experiment in Figure 1.5. Details regarding the design of the mobile robot and the experiments conducted with it are provided in Chapter 6.

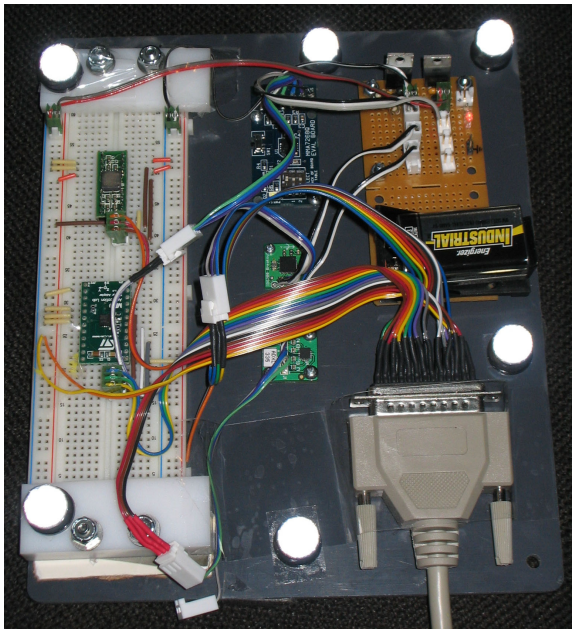


Figure 1.1: Evaluation board collection used for preliminary experiments.

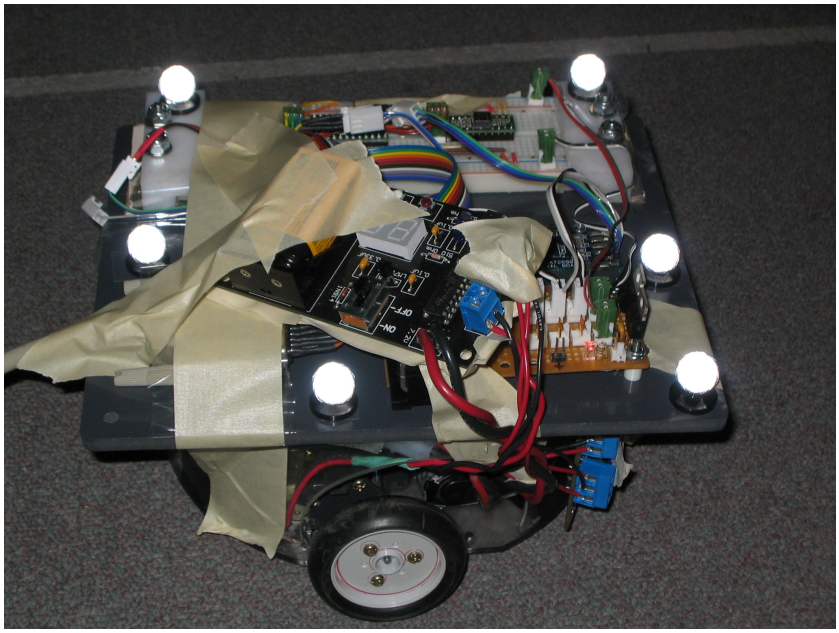


Figure 1.2: Evaluation board collection mounted on a repurposed mobile robot.

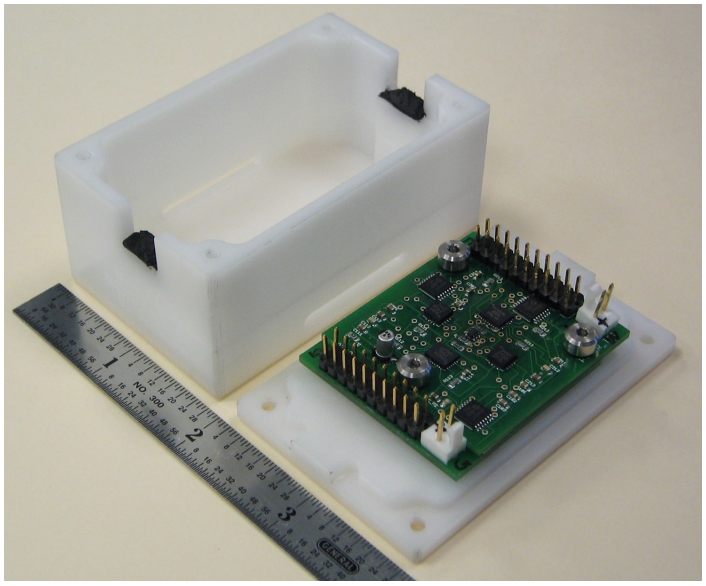


Figure 1.3: Custom IMU designed for this research. A ruler with major divisions in inches is also shown for scale.

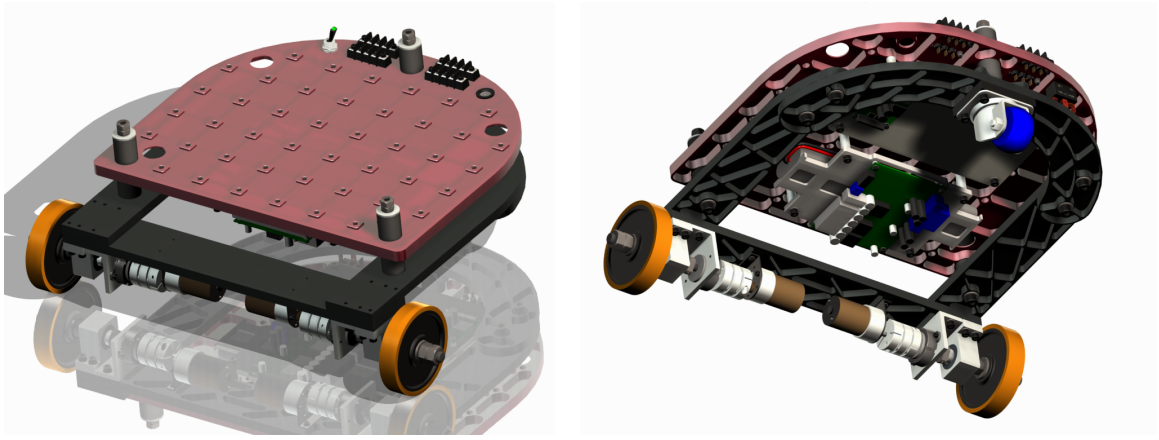


Figure 1.4: Renderings of the mobile robot designed for inertial measurement experiments.

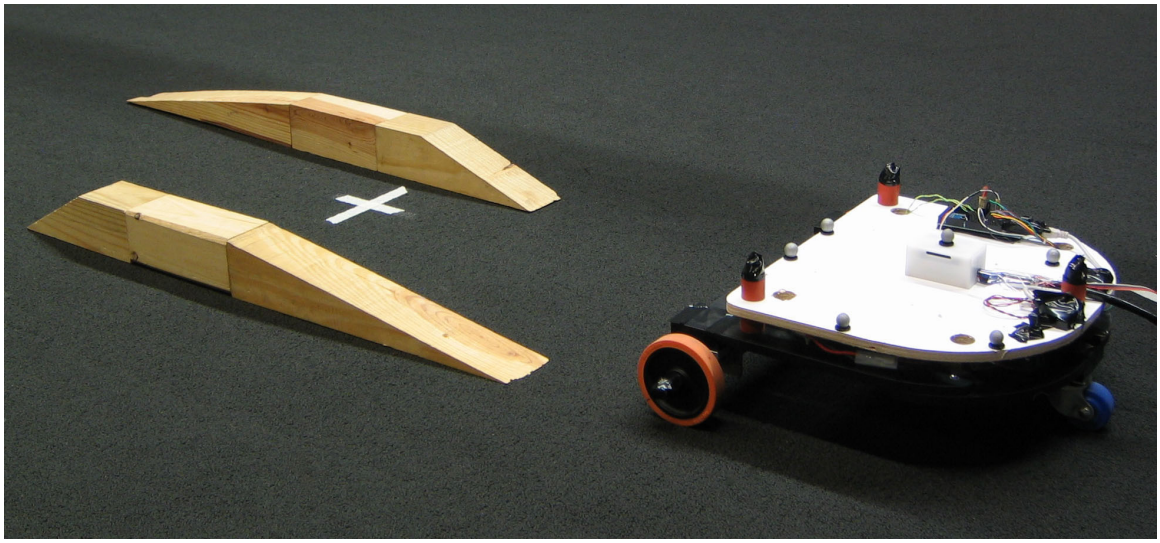


Figure 1.5: Constructed robot being used for motion tracking experiments.

Third, the hardware necessary for the implementation of a novel calibration technique was designed and manufactured. This included the case of the inertial measurement unit, two support fixtures, and a plate providing reference surfaces. These items are shown together in Figure 1.6. Information regarding this equipment and the calibration technique are given in Chapters 3 and 4.

Fourth, a significant amount of work was contributed to the design of a machine that, once constructed, will allow the calibration process and tests of spatial motion tracking algorithms to be automated and performed with greater accuracy. A rendering showing the progress made on the design of this machine is shown in Figure 1.7.

1.3.2 Calibration technique

A new calibration technique was developed specifically for MEMS inertial sensors to allow the calibration of complex sensor models. These models include axis misalignment, scale factor nonlinearities, anisotropic effects, and gyro specific-force sensitivity. The observation equations are formulated such that least-squares regression can be used to

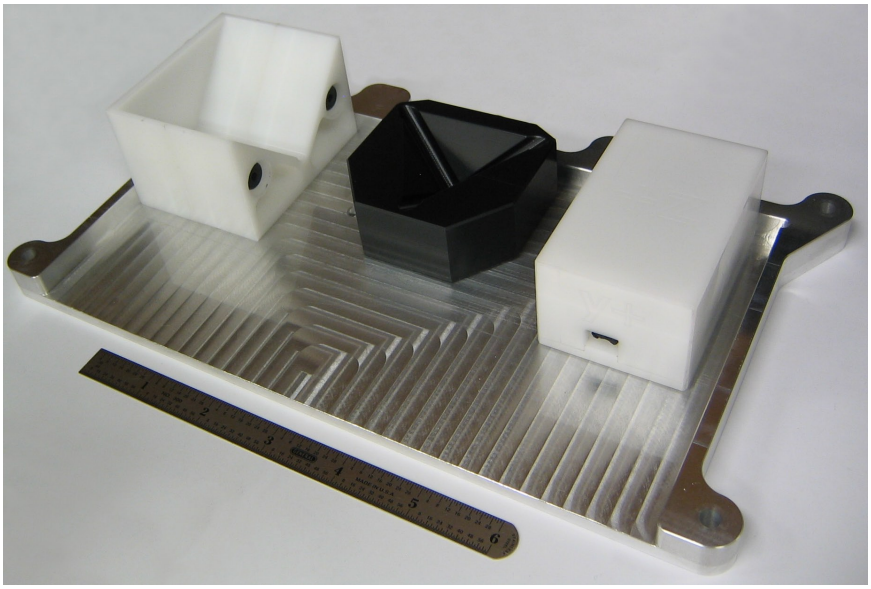


Figure 1.6: Hardware developed to implement the calibration technique. A six-inch ruler is also shown for scale.

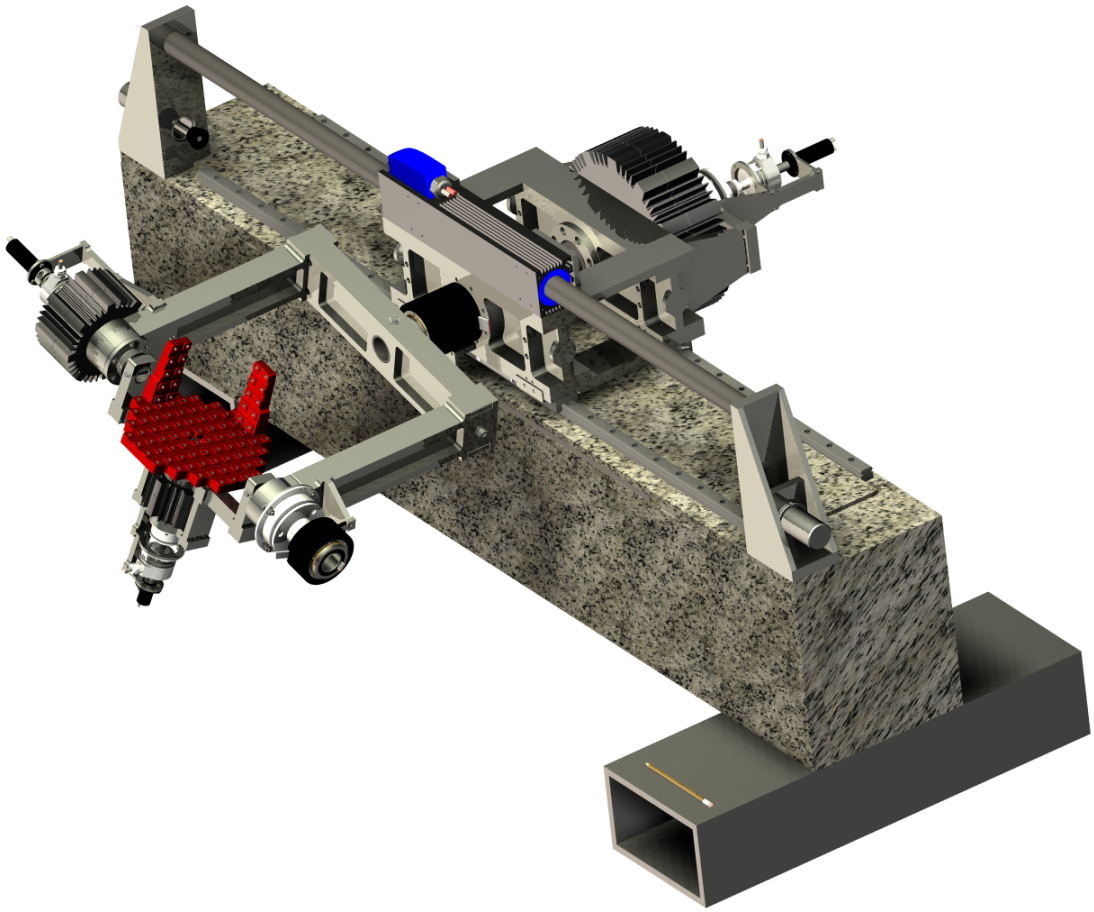


Figure 1.7: Rendering of the design for the automated calibration and testing machine.

estimate the model parameters. This makes the numerical implementation of the calibration method simpler and accessible to a wider audience by avoiding the nonlinear estimation techniques required in many alternative methods.

An inexpensive method of implementing the calibration technique was also developed. It uses manual excitation of the sensors and motion constraints provided by a plate with reference surfaces, the geometry of the IMU case, and simple support fixtures. This avoids the prohibitive cost of precision, automated equipment that has traditionally been required to implement techniques capable of calibrating complex sensor models. Experimental results demonstrate that utilizing these advanced sensor models provides significant improvements in the accuracy of the resulting specific force and angular rate estimates compared to the linear models employed in calibration techniques that have previously been developed for use with MEMS inertial sensors.

1.3.3 Error reduction and navigation algorithms

Algorithms to process the outputs from an inertial measurement unit and identify periods where corrections can be made to reduce error growth rates were developed. These algorithms use easily identifiable characteristics of the sensor outputs to recognize periods where no measurable inputs are present and the sensor bias levels can be updated. This allows integration errors to be halted and error growth rates to be reset, drastically improving navigation results.

These algorithms were first developed and demonstrated for one-dimensional motion tracking using an accelerometer. An extension to full, three-dimensional motion tracking was also developed and tested in an application with a mobile robot on uneven terrain. In these tests, centimeter-level positioning errors were achieved over time spans of up to 30 s given reasonably frequent pauses in the robot's motion. This demonstrates that inertial measurement using MEMS sensors is, indeed, capable of providing accurate motion tracking information over periods of time long enough to be useful in many applications.

This work has provided the University of Utah with the equipment and techniques necessary to participate in state-of-the-art testing and development of MEMS inertial measurement technology. In addition, the sensor-level error compensation techniques were developed with an emphasis on creating algorithms that rely on inherent sensor characteristics rather than application-specific heuristics. This allows the resulting inertial navigation solutions to be used in a wide variety of applications.

1.4 Document organization

The remainder of this document is organized as follows. Chapter 2 provides additional background for the project by presenting a more detailed overview of existing work in the field of human-scale inertial measurement. The mathematical framework of the calibration technique developed as part of this research is presented in Chapter 3 along with contextual background through a review of previously existing calibration techniques for inertial measurement units. The implementation of the new calibration technique along with simulated and experimental results follows in Chapter 4. The one-dimensional bias tracking and error correction algorithms are described and demonstrated in Chapter 5. Chapter 6 then presents the extension of these algorithms for three-dimensional motion tracking with experimental results for a mobile robotics application. Finally, a conclusion and summary of the contributions from this work are given in Chapter 7.

CHAPTER 2

PRIOR WORK IN HUMAN-SCALE INERTIAL MEASUREMENT

2.1 General description of prior research

In the early 1990s, one of the first human-scale applications of inertial navigation using MEMS-based inertial sensors was presented in the work of Barshan and Durrant-Whyte in the area of mobile robots [13–15]. Around the same time, Glynn filed a patent for a computer input device based on inertial measurement [28]. Since then, many other researchers have explored this idea of an input device that responds to spatial motion [20, 27–32]. Small, unmanned aerial and underwater vehicles have also been the subject of much research in inertial measurement (see [10–12] for just a few examples) with a group at Brigham Young University developing a commercial autopilot package for micro-aerial vehicles that is aided by MEMS-based inertial navigation [33].

One of the earliest applications of human motion tracking using these miniaturized inertial sensors is found in Foxlin's development of an attitude and heading reference system for tracking head motion in virtual reality applications [16, 17]. After this, other human motion tracking applications began to be explored. To date, these primarily fall into two categories: (1) studies that track only body segment orientation and use a kinematic model to relate this to joint angles and, thereby, limb and/or torso position [21–23, 34]; and (2) studies that track foot motion through true inertial navigation [18, 19, 35].

After reviewing these different studies, the following general observations about the current capabilities and limitations of human-scale motion tracking can be made. These will be validated through a more detailed discussion of representative studies in Section 2.2.

- Without the aid of outside references or application-specific simplifications of some kind, MEMS-based inertial navigation is currently able to provide useful position estimates for only a few seconds. This is primarily attributed to white noise and instability in the parameters of these sensors, along with the unfortunate, but unavoidable interaction between orientation error and inaccurate gravity cancellation in the accelerometer readings [13–15, 18, 35–37].
- The combination of a three-axis accelerometer acting as a tilt sensor and a three-axis gyro acting as an attitude and heading reference system can provide drift-free pitch and roll orientation estimates for unlimited amounts of time provided that there are pauses in the motion from time to time [16, 38]. Between pauses, the error growth rate is at worst equal to the drift rate of an unaided gyro-based system and can be much better if accelerations are low, allowing better estimation of the gravity vector detected by the accelerometers.
- While inertial sensors cannot provide drift-free yaw estimates on their own, a three-axis magnetometer can be utilized to overcome this limitation [17, 18, 39]. Unfortunately, using these magnetic sensors is greatly complicated by the need to compensate for disturbances in the Earth's magnetic field whenever objects containing ferrous materials or other magnetic influences are nearby [34].
- If the motion to be tracked is composed of short, discrete motion segments with intervening pauses, then the integration of the inertial sensor outputs can be performed over these short segments of motion and the error growth rate can be reset with each pause. This is accomplished using methods commonly referred to as zero-velocity updates [18, 35, 40, 41] or zero-velocity compensation [20]. Using these techniques, and assuming each motion segment is comparable in duration, the overall error growth rate becomes linear in the number of motion segments executed, rather than growing at an ever-increasing rate with time.

2.2 Discussion of representative studies

2.2.1 Unaided inertial navigation

In [13–15] the authors investigate the possibility of using inertial navigation to provide motion tracking for a mobile robot. They present a sensor model that accounts for the temperature-induced bias drift of the inertial sensors as they warm up and reach a steady-state operating temperature. Using this model, they perform a linear motion experiment using a mobile robot test platform. In these tests, reasonable results are obtained, but the drift rate in the position estimate is observed to be 60–80 cm in 10 s. This strong drift rate leads to rapid divergence between the true and estimated position of the robot.

While these tests were performed 15 years ago, there still do not appear to be any significant improvements in the results obtained through open-loop inertial navigation over periods of time lasting more than a few seconds. Even results published last year for a preliminary test of unaided inertial measurement with a mobile robot [42] show very comparable results. Therefore, inertial measurement in mobile robot applications has almost exclusively been used as an aiding system to odometry [43], GPS [44], or visual [45] motion tracking systems.

Another study aimed at evaluating unaided inertial measurement for use with mobile robots is presented in [36]. Here, the authors illustrate the sensitivity of the integration results on accurate knowledge of the sensor bias level. By manually tuning the bias level of the sensor, they were able to obtain useful results over periods of approximately 20 s. However, without this manual tuning, position error drift rates comparable to those in [13–15, 42] are observed.

In these studies, the large error growth rates are attributed to sensor bias level uncertainty and instability. It should be noted, however, that cross-axis effects, nonlinearities, misalignments, temperature sensitivity, and other effects that also cause shifts in the voltage output of the sensors will contribute to error growth in the same way as bias drift. In [46,47], the authors demonstrate that cross-axis effects and nonlinearities in MEMS inertial sensors can have a magnitude ten times larger than bias drift over time. They report shifts

in the output of a MEMS accelerometer on the order of 100 mV due to nonlinearities and cross-axis effects during tumble testing whereas the bias drift observed in a 12 h static test was only 10 mV. These results suggest that accurate compensation for this type of sensor-level error sources could significantly improve the accuracy of inertial navigation using these sensors.

For three-dimensional navigation, errors in the orientation estimates obtained from the gyros also contribute significantly to errors in position estimates. Simulated and experimental results in [37] show that inaccurate cancellation of gravity due to orientation errors becomes the dominant source of drift in the position estimates after only a few seconds for MEMS inertial measurement units. Unlike bias drift in the accelerometers, which is bounded within relatively small limits, as orientation errors grow in time, so does the component of gravity that is erroneously interpreted as kinematic acceleration. This relationship is discussed further in Section 2.2.2 below.

2.2.2 Limiting the length of integration periods

Assuming a constant error in the estimate of the bias level of an accelerometer axis, and a linearly growing error in gravity compensation due to orientation error growth from gyro drift, the resulting error in the position estimate will be given by:

$$\delta_x = \iint (a_b + kt)\, dt = \frac{1}{2} a_b t^2 + \frac{1}{6} k t^3 \tag{2.1}$$

where δ_x is the position error, a_b denotes the acceleration corresponding to the constant accelerometer bias error, k represents the growth rate of the acceleration error due to gravity compensation errors, and t is time. While the relationship between orientation error and gravity compensation errors is in actuality sinusoidal, this approximation is accurate for small orientation errors.

Given the t^2 and t^3 terms present in (2.1), the risk of performing open-loop integration

for long periods of time becomes clear. As an example, assume a somewhat conservative bias error of $a_b = 0.025$ m/s^2. In addition, assume that error in the orientation estimate is developing at the conservative rate of 0.1°/s, giving $k = 0.017$ m/s^2. The resulting evolution of the components of δ_x is illustrated in Table 2.1. This simple example demonstrates the drastic effects of a cubic rate of error growth. After only 1 min of integration, there is more than half a kilometer of error in the position estimate. In fact, if an application required less than 10 cm of error for the position estimate to be useful, then this limit would be reached after just over 2 s of integration.

The preceding example illustrates why limiting the length of integration periods is such a powerful tool. In [18, 35], Foxlin presents a foot tracking system for pedestrians that utilizes inertial navigation and can operate with or without the aid of GPS signals. The algorithm is carefully developed to ensure that integration only occurs while the foot is in the air, a period that is typically less than 0.5 s. While the foot is on the ground, it is stationary and the algorithm detects this stance phase and uses it to provide information from zero-velocity updates to an extended Kalman filter. Since these papers describe proprietary technology, they do not provide detailed information about the information utilized during the zero-velocity updates. However, it is likely that the algorithm capitalizes on the fact that velocity must be zero while the foot is planted and accelerations are also zero, allowing sensor bias levels to be updated and the accelerometers to be used as tilt sensors to help correct the orientation estimate. Updating the orientation estimate in this manner will be discussed further in Section 2.2.3.

Table 2.1: Example of position estimate error growth for a representative accelerometer bias error and orientation error growth rate.

t (s)	0	0.25	0.5	1.0	2.0	5.0	10	30	60
$\frac{1}{2}a_b t^2$ (m)	0	0.001	0.003	0.0125	0.05	0.31	1.3	11	45
$\frac{1}{6}kt^3$ (m)	0	0.00004	0.0004	0.003	0.023	0.36	2.9	77	620
δ_x (m)	0	0.001	0.003	0.015	0.073	0.67	4.1	88	660

By limiting periods of integration to around 0.5 s, position error on the order of only 3 mm is expected to accumulate with each step according to the example presented in Table 2.1. Assuming strides are taken with a frequency of 1 Hz, after 1 min, 60 steps will be taken giving an expected error of $60 \cdot 3$ mm = 0.180 m. While still substantial, this is a significant improvement relative to the potential 660 m of error that would be expected without partitioning the motion to reduce the duration of the integration periods. Bamberg et al. [19] used a similar technique to achieve errors of 7.4 ± 13.6 cm using a MEMS inertial measurement unit, as determined from 315 strides using a vision-based clinical motion tracking system as the truth reference.

Another example of a successful application of inertial navigation using short integration periods is the development of a human-computer interface based on spatial motion tracking presented in [20, 29, 32]. In these studies, the interface device is equipped with a triaxial set of accelerometers but no gyros. The device is used by quickly writing one of a predefined set of characters in the air. These motions do not involve significant changes in the orientation of the device and they last for less than a second. Because of this, the authors are able to define acceleration thresholds that can be used to detect the start and stop of each motion. They use the accelerometers as tilt sensors to estimate roll and pitch angles before and after each motion and linearly interpolate the orientation between these values, neglecting any changes in the yaw angle. Using these orientation estimates, they integrate the accelerometer readings and produce a three-dimensional trajectory that is then projected onto a fitted plane to produce the character. This is then processed using standard character recognition techniques.

Despite the lack of any gyros in this device and the simplifying assumptions made in the data reduction, they are able to obtain a 95% character recognition rate. The key element in their success is the partitioning of the motion into rest periods and short, meaningful strokes where integration is actually performed. A similar method is presented in [27] where the spatial input device is equipped with a gyro triad. This improves the

orientation estimate during the stroke and, consequently, the recognition rate for most characters.

2.2.3 Attitude and heading reference systems

As a final area of discussion, consider the use of MEMS-based inertial sensors to implement an attitude and heading reference system used to track the orientation of a body. The development of such a system is described in great detail in [16]. In this study and others like it [17, 21, 34, 38, 39], it is experimentally shown that despite the drift and uncertainties in their bias level, accelerometers can be used as high-bandwidth tilt sensors that provide sufficiently accurate results to correct and limit pitch and roll orientation errors to around 2° during pauses in the body's motion. This allows drift due to inaccuracies in the integrated gyro outputs to be corrected.

As with the motion tracking methods discussed in Section 2.2.2 above, the success of this tilt sensing technique depends on relatively frequent pauses in the motion and the amount of error growth between pauses increases with the duration of the motion. However, there are many practical applications where this limitation is perfectly acceptable. Furthermore, unlike error in the position estimate, accumulated error in the pitch and roll components of the orientation estimate can be eliminated during each pause since Earth's gravity vector provides a convenient absolute direction reference. One other advantage for orientation tracking is that the error growth rate during periods of motion depends only on the accuracy of the gyro outputs. Therefore, it is expected to be only linear in time rather than cubic, as for the position estimate.

With gyros and accelerometers alone, yaw angle errors cannot be corrected. Therefore, many MEMS-based attitude and heading reference systems also incorporate a magnetometer. Doing this allows the Earth's magnetic field to provide a second absolute direction reference that can be used to correct yaw errors. Compared to accelerometers, the magnetometers offer the distinct advantage of not being influenced by the body's motion.

Unfortunately, the Earth's magnetic field is relatively weak and is easily disrupted by nearby ferrous materials or other magnetic fields. A simple illustration of this problem can be performed by holding a ferrous object or an electronic device near a compass. The needle will report north to be directly toward the object once they are brought within a few inches of each other. Both [34] and [18] show that sophisticated compensation techniques are necessary to obtain satisfactory results in the presence of such disturbances. Seeing as these disturbances are present in most modern environments, this means that adding magnetometers to an inertial measurement unit creates a large increase in the complexity of the processing algorithms.

2.3 Summary of prior work

The preceding discussion has shown that while the implementation of MEMS-based inertial navigation systems still presents daunting obstacles, they have been successfully used in several practical applications. Unfortunately, to date, the techniques utilized in these studies have depended heavily on application-specific simplifications, empirically-derived thresholds, and other heuristics that limit the versatility of the algorithms developed for a particular application. There has been some exploration into the possibility of improving the results of inertial navigation using MEMS sensors through improved sensor-level modeling and error compensation; however, this work is still in the very preliminary stages. A significant obstacle in its development has been the limited availability of the precision equipment traditionally required to perform this type of detailed calibration.

CHAPTER 3

A POWERFUL AND INEXPENSIVE METHOD FOR CALIBRATING AN INERTIAL MEASUREMENT UNIT: BACKGROUND AND THEORY

Eric Allen Johnson and Stacy J. Morris Bamberg

Department of Mechanical Engineering, University of Utah

Salt Lake City, UT 84112

The contents of this chapter have been submitted to *Sensors and Actuators A: Physical* for review. Formatting and other minor modifications have been applied for consistency with the remainder of this dissertation.

3.1 Abstract

MEMS inertial sensors are increasingly popular and capable, but require calibration to obtain accurate outputs. This chapter provides a categorization and review of calibration techniques targeting MEMS inertial sensors and discusses outstanding limitations of existing techniques. A new calibration technique that addresses key limitations is then presented. The method utilizes geometric constraints that can be enforced using simple fixtures. A novel mathematical approach compares integrated sensor outputs with known linear and angular displacements to allow complex models including nonlinearities, anisotropic sensitivity, and gyro specific-force sensitivity to be calibrated using standard linear regression. The method can be implemented using manual excitation of the sensors and does not require known angular rate inputs or precise alignment of the sensors to an external field.

3.2 Introduction

Since their introduction, the price of inertial sensors based on micro-electromechanical systems (MEMS) has steadily decreased while their performance has steadily increased [1, 6, 8, 9]. They have also benefited from continuous size reductions and increased levels of integration. Techniques to integrate MEMS accelerometers and angular rate sensors, or gyros, into a single package at the wafer level are currently being researched [48]. The development of mass-produced, commercial sensors utilizing such techniques is underway [49, 50] and will soon make a six-degree-of-freedom (triaxial accelerometer and triaxial gyro) MEMS inertial measurement unit (IMU) available in a single integrated sensor package that occupies only 14.4 mm^3 [50].

These advances have inspired the use of MEMS inertial sensors in a wide variety of applications, examples of which can be found in [51]. Custom sensor packages are often developed to minimize size and cost. The accuracy of such devices is limited by the inability to ensure exact alignment of the sensor axes during construction and the

inherent variability in the parameters of inexpensive MEMS inertial sensors. Overcoming these limitations to improve the accuracy of the IMU outputs requires calibration of the assembled unit.

In applications involving navigation-grade sensors, specialized, precision equipment is used for their calibration [1, 2, 52]. For researchers and clinicians utilizing MEMS inertial sensors, they are usually a means to an end, not the main subject of study. Therefore, the expense of purchasing, installing, maintaining, and operating precision calibration equipment would be unjustified. This is especially true considering that these costs typically exceed the cost of the MEMS inertial sensors by several orders of magnitude.

In some applications, particularly in the medical field, obstacles in calibrating the sensors and inaccurate results from integrating uncalibrated outputs has limited the use of MEMS inertial sensors to peak detection and pattern recognition using raw outputs. In applications where increased accuracy is requisite, the prohibitive cost of specialized calibration equipment has motivated creative adaptations such as record players [53] and bicycle wheels [51] substituting for rate tables. It has also inspired the development of alternative calibration techniques. These vary significantly in approach, the complexity of the sensor models addressed, and the complexity of the equipment and mathematics required. In general, reductions in the complexity of required equipment have been accompanied by increased complexity in the solution methods. Techniques appropriate for MEMS sensors have also generally been limited to simple linear sensor models sometimes accounting for axis misalignment within and between sensor triads.

In this chapter, a new calibration technique is presented that allows complex sensor models to be calibrated including nonlinearities, anisotropic effects, and gyro specific-force sensitivity. A novel mathematical approach allows these complex sensor models to be calibrated using simple linear regression. This makes the computational implementation of the technique more accessible and allows the use of well-established techniques associated with least-squares analysis to assess the quality of the calibration results. Chapter 4 shows

how the technique can be implemented using manual excitation of the sensor axes along with simple, inexpensive fixtures to control the orientation of the IMU.

To provide background for this method, existing calibration techniques are categorized and summarized in Section 3.3. Sufficient information is provided to assist in identifying methods that may be well suited for particular applications. Section 3.4 defines the sensor models used in the new calibration technique, which is presented in Section 3.5. Methods of assessing the calibration quality are discussed in Section 3.6 and Section 3.7 examines key assumptions employed in the calibration technique and the flexibility it offers to adapt the models to particular sensors. Finally, concluding remarks are given in Section 3.8.

3.3 Overview of existing calibration methods

The existing IMU calibration techniques discussed in this section are grouped into the following four categories.

1. Calibration techniques that require precision equipment to implement (Section 3.3.1). These methods rely on precise knowledge of the orientation and motion of the IMU allowing the inputs due to gravitational acceleration, Earth's rotation rate (when it can be detected by the gyros), and the applied motion to be used as reference values in the calibration. They allow the most complex sensor models to be calibrated, but the cost of the requisite equipment is prohibitive for researchers using MEMS sensors.
2. Magnitude-based calibration techniques (Section 3.3.2) that relax the requirement of precisely controlling the IMU's orientation by using vector magnitudes in the cost functions for the calibration equations. On their own, these techniques do not allow accelerometer and gyro triads to be aligned to each other or to external frames because magnitudes do not provide information about the direction of applied inputs. Section 3.3.2.1 discusses recently developed extensions that begin with a magnitude-based calibration to determine all intratriad calibration parameters and then use rotations about fixed axes to determine their relative alignment or their alignment to an external frame.

3. Geometry-aided calibration techniques (Section 3.3.3) that use the geometry of the IMU's housing and other references to assist in the calibration. In these methods, the geometry of the objects involved control the IMU's orientation rather than precision positioning devices as in traditional laboratory techniques. While they are not common, these methods are given special attention because the technique presented in this chapter uses a similar approach.
4. In-use calibration techniques (Section 3.3.4) that are designed to calibrate some or all of the sensor model parameters while the IMU is collecting data in the intended application. This section focuses on techniques applicable to unaided inertial measurement as that is typical for MEMS IMU applications. While these methods are typically limited in the complexity of the sensor model that can be calibrated, they directly address concerns with in-run and run-to-run stability of commercial-grade MEMS inertial sensors.

Section 3.3.5 concludes this section by summarizing specific limitations of existing calibration techniques designed for use with MEMS inertial sensors that are addressed by the new method presented in this work.

3.3.1 Calibration techniques requiring precision equipment

Chatfield [2, Ch. 5] describes a traditional laboratory calibration technique for navigation-grade IMUs mounted either in a gimbaled platform or on a three-axis rate table. Accurate leveling and precise control of orientation allow gravity to be used as the acceleration input. The stable platform torquers or rate table axes are used to provide angular rate inputs that are precisely controlled and aligned. An 18-observation rotation schedule is recommended with the rotation axes to travel between each orientation specified. This provides sufficient information to calibrate an accelerometer model that includes intratriad misalignment, scale factor, scale factor nonlinearity, and bias correction. The gyro model includes bias, scale factor, misalignment relative to the accelerometer frame, and linear, squared, and anisotropic specific-force sensitivity.

Since the orthogonalized accelerometer frame is taken as the reference frame for the IMU, a magnitude-based solution technique is recommended for the accelerometers. This makes the result insensitive to small orientation errors. The error equations are linearized and only first-order terms are retained to allow a least squares solution for the model parameters. For the gyros, both magnitude- and vector-based solutions are presented. The magnitude-based solution leaves the parameters required to align the gyro triad to the accelerometer frame unobservable, but does not require precise knowledge of the direction of applied angular rates or accelerations. To calibrate the full model, the vector form must be used. This requires precise knowledge of the direction and magnitude of the applied angular rates and the apparent gravity vector.

As an alternative to this batch processing method, Chatfield notes that Kalman filters are often used to dynamically estimate the parameters during the platform rotations. An example of this technique is presented by Grewal et al. [5], where an extended Kalman filter with 63 states is implemented to calibrate accelerometer and gyro models of similar complexity to those used by Chatfield [2]. This technique also assumes that the orientation of the IMU can be precisely controlled using either a gimbaled platform or a three-axis rate table. Careful design of the rotation trajectory is necessary to ensure that all of the states will be observable.

Temperature compensation is also a concern to maximize the accuracy of the IMU outputs. Two approaches are generally taken [1, Ch. 8]. Either the temperature of the IMU can be controlled during use, or calibration of the IMU can be performed over a range of temperatures allowing some or all of the sensor parameters to be modeled as functions of temperature. For MEMS inertial sensor applications, controlling the temperature is usually impractical due to the required increase in power consumption. Therefore, existing examples of temperature compensation applied to MEMS inertial sensors involve recalibrating the sensors over a range of temperatures. Automating this process for even the simplest sensor models requires a temperature-controlled chamber with at least a single-axis rotary

table with the axis perpendicular to gravity. Examples of results for MEMS inertial sensors using this type of equipment can be found in [54–58].

As mentioned previously, the cost of the equipment required to implement the techniques described in this section cannot be justified for end-users of MEMS inertial sensors. Alternate calibration techniques based on precise measurements have been developed around different types of specialized equipment available in particular settings. In [59], a three-axis vibration table is used to excite a triaxial accelerometer and the motion is measured using laser interferometers. An optical tracking system is used to provide reference acceleration and angular rate estimates to calibrate an IMU in [60]. A six-degree-of-freedom robotic manipulator is used to control the orientation and calibrate triaxial accelerometers and magnetometers in [61]. In [62] an optical cube and beam collimators are used to precisely orient and calibrate misalignment for an attitude and heading reference system using accelerometers and magnetometers. In [58] and [63], special fixtures are used with a single-axis rate table to allow the sensors to be placed in precise orientations relative to the rotation axis. A triaxial gyro cluster is characterized in [58] and [63] calibrates a full IMU including centripetal acceleration as one of the inputs for the accelerometers to allow them to be calibrated over a wider range of inputs than is possible using gravity alone. Finally, [64] developed a custom, manually-actuated, three-axis gimbal along with a technique to calibrate an IMU using the outputs of encoders on each joint.

3.3.2 Magnitude-based calibration techniques

Magnitude-based, or multiposition, calibration methods are attractive for MEMS inertial sensors since they do not require careful alignment with the gravity vector and precise orientation control. An early example of magnitude-based calibration for MEMS accelerometers is given in [65] for a medical activity monitoring application. A simple linear sensor model using only bias and scale factor is used along with a nonlinear least squares estimation technique to obtain the parameter estimates. The technique presented

in [66] uses this same basic sensor model, but contributes an iterative solution technique designed specifically for the model. It is very robust to errors in the initial parameter estimates and converges rapidly.

In his thesis, Shin [67] presents a sensor model and solution technique that accounts for intratriad misalignment in addition to the bias and scale factor. He suggests that the technique could also be applied to MEMS gyros if a stronger rotation input than is provided by Earth's rotation rate could be applied. This is done in [68] where the same research group uses a single-axis rate table to allow both the gyros and accelerometers in a commercial and a custom-built MEMS IMU to be calibrated. They use a nonlinear least squares solution technique that requires a preliminary calibration to obtain sufficiently accurate initial values for the parameter estimates to ensure correct convergence.

Another group of researchers developed a magnitude-based calibration approach in the framework of ellipsoid fitting [69,70]. Their sensor model also addresses bias, scale factor, and intratriad misalignment and the solution method trades the complexity of iterative nonlinear least squares for complex algebraic manipulations. Their method allows intermediate variables describing the equation of a generalized ellipsoid to be found using linear regression. The original sensor model parameters may then be recovered algebraically.

Skog and Händel [71] also present a magnitude-based calibration technique accounting for bias, scale factor, and intratriad misalignment utilizing a nonlinear least squares solution. They provide an analysis demonstrating that the mean square error in the parameter estimates obtained using their procedure is near the Cramér-Rao lower bound. Due to the lack of a rate table, they are unable to demonstrate calibration results for gyros, but this is done using their technique in [53] where the authors adapt a record player for use as a rate table. The variances of the calibrated parameter estimates reinforce the need for precision-controlled rate tables for this approach.

3.3.2.1 Extensions to allow intertriad alignment

As noted, one drawback of the exclusive use of magnitude-based calibration techniques is the loss of directional information needed to calibrate intertriad alignment. A two-step technique to address this is presented by Včelák et al. [72]. Following magnitude-based calibration of accelerometer and magnetometer triads, they use a fixture resembling a theodolite to rotate the sensors about two orthogonal axes. Due to misalignment between the sensor axes and the rotation axes, a sinusoidal output is observed and misalignment parameters are fit to eliminate this variation, effectively aligning the triads to the rotation axes. Jurman et al. [39] apply this technique to calibrate accelerometers with the method from one of the geometric techniques [73] presented in Section 3.3.3 to calibrate gyros.

Bonnet et al. [74] consolidate ideas from several magnitude-based calibration techniques and present a thorough analysis in the framework of ellipsoid fitting. They also formalize the external alignment technique used in [72] by noting that the projection of a reference vector (gravity for accelerometers or Earth's magnetic field for magnetometers) onto a fixed rotation axis is constant. They show how this principle can be used to identify the rotation axis in the sensor frame using singular value decomposition and principle component analysis. By rotating about each axis of an orthogonal reference frame, the transformation between the sensor and reference frame is obtained. A method of compensating for rotations about axes that are not perfectly orthogonal and adaptations to use the technique for gyros are also described.

Zhang et al. [52] also summarizes and formalizes ideas from previous magnitude-based calibration methods. They specifically seek to address the issues of intertriad alignment, and optimizing the set of observations used in the magnitude-based calibration step. Intertriad alignment is obtained by identifying two linearly independent vectors in both the accelerometer and gyro frames. These two vectors and their cross product allow the relative orientation of the sensor triads to be determined. To identify the vectors, they note that during a rotation about a fixed axis, the gyros provide a direct estimate of that axis. The

same axis is identified in the accelerometer frame by taking the dot product of the rotation axis with the accelerometer readings. Doing this gives a constant projection of gravity onto the rotation axis and cancels out any centripetal acceleration due to the rotation since, by definition, it is perpendicular to the axis of rotation. Eigenvector analysis then allows the rotation axis to be identified. They also perform a sensitivity analysis to determine an optimal set of observations for the magnitude-based portion of the calibration. The results show that the cost function's sensitivity to the parameters in their model is proportional to either the reading along individual sensor axes or the product of the readings along two separate axes. This suggests aligning the IMU axes along the reference vector and at 45° orientations between those positions, which is an intuitive result that agrees with the rotation schedule presented by Chatfield [2].

3.3.3 Geometry-aided calibration techniques

The method presented by Ferraris et al. [73] is an oft-cited example of using geometry to facilitate the calibration of an IMU. A leveled, flat surface and a reference straight edge are used to calibrate an IMU housed in a rectangular prism. The authors calibrate an accelerometer model that includes bias, scale factor, and misalignment relative to the frame provided by the IMU's housing. This is done using data collected from the six orientations obtained by placing each face of the housing down on the leveled surface. The model parameters are then found through algebraic and matrix operations using the data from these observations. The gyro model includes a time-varying bias, scale factor, misalignment, and linear specific-force sensitivity. It is calibrated using the data collected for the accelerations plus a sequence of rotations performed while keeping one of the IMU housing's faces in contact with the flat surface. This fixes the rotation axis and rotations of exactly 360° are performed by starting and ending with the same side of the housing against the straight edge. The data from these rotations are integrated allowing the rotation angles to be used to calibrate the scale factor and misalignment parameters. This eliminates the

need to measure the actual angular rates during the motions. A method of using information from a coordinate measuring machine to compensate for errors in the orthogonality of the housing's faces is also presented.

Another geometry-aided calibration technique is presented by Hwangbo and Kanade [75]. It is based on a factorization technique developed to separate the shape of objects from camera motion in computer vision. Applied to IMU calibration, they use the factorization to separate data into components due to the sensor model (shape) and the applied orientations and rotations (motion). The sensor models used include bias, scale factor, and misalignment relative to an external frame for both the accelerometers and gyros. The technique requires knowledge of relationships between the applied motions, such as orthogonality and parallelism, rather than their precise directions. By mounting the IMU on a universal angle block that would typically find use as a reference in a machine shop, they collect data with known motion relationships by placing the block in different orientations and rotating it through known angles on a flat surface with a straight edge reference. The geometrical constraints used are effectively identical to those of Ferraris et al. [73], but the factorization-based solution approach is, of course, quite different.

3.3.4 In-use calibration techniques

Extended Kalman filters are traditionally used for in-use IMU calibration and alignment. Examples of these methods are given in [1–3, 76]. Unfortunately, they are difficult to apply to MEMS IMUs because, by nature, Kalman filters depend on two independent estimates of a process. For traditional applications, gyro-compassing, a gimbaled platform, or GPS provide the second estimate, but these are generally unavailable or undesirable in MEMS IMU applications. Because of this, in-use calibration techniques for MEMS inertial sensors generally depend on collecting data during static periods when the accelerometers can be used as tilt sensors to estimate the gravity vector. This section provides examples of such techniques.

The magnitude-based calibration technique of [65] is presented in a framework that allows the calibration to be performed during use. A method of detecting static periods in the data is presented and used to collect data points that can be used in the sensor calibration. Since the intended application is medical activity monitoring, they compare two potential orientations of the accelerometers on the patient and determine the minimum range of tilt that must be imparted before accurate parameter estimates can be obtained. The method considers only accelerometers with a very simple sensor model—only bias and scale factor are considered—and "static" data obtained during use is inherently more noisy than would be obtained in a laboratory calibration, but the method is shown to be viable and to give good results for their application.

A slight extension of this work is given in [77]. Here, both accelerometers and gyros are used and a preliminary calibration provides initial values for the linear sensor models and misalignment parameters. During use, a different static period detection method is used. Gyro biases are updated directly via averaging during static periods and data points are collected to use in the in-run accelerometer calibration method.

Fong et al. [78] present a very useful method that allows the IMU to be fully calibrated during use without any external equipment. It is demonstrated both for laboratory calibration and in-use calibration with the IMU mounted on a user's hand and head. They use gravity as the reference to calibrate accelerometers using a magnitude-based calibration that includes bias, scale factor, and misalignment. Data points are collected during static periods and the parameters are estimated using a general-purpose nonlinear estimation technique to minimize the difference between the observed and reference gravity magnitudes. To calibrate the gyros, they note that the direction of gravity as estimated by integrating the gyro outputs to track the IMU's orientation should match estimates of its direction obtained by using the accelerometers as tilt sensors during static periods. Hence the cost function for the gyro calibration becomes the magnitude of the difference between these two estimates of the gravity vector. The same nonlinear estimator is used and allows gyro bias, scale

factor, and misalignment relative to the accelerometer triad to be calibrated.

The most prominent difficulties they encountered in implementing the method were: (1) identifying static periods from which robust estimates of the gravity vector can be obtained using the accelerometers and (2) ensuring sufficient variation in the data to allow the nonlinear estimators to converge on a correct solution. Both problems were more prominent in the head-mounted tests where the IMU does not remain as still during static periods and is only subjected to relatively small changes in orientation. Another concern is that the accuracy of the accelerometer's tilt sensing limits the accuracy of the reference orientation measurements used to calibrate the gyros. Redundancy in the data may help average out these errors.

3.3.5 Limitations of existing calibration techniques

Higher-order effects such as scale factor nonlinearity, anisotropic sensitivities, and specific-force sensitivity in the gyro response are expected to be greater for inexpensive MEMS inertial sensors than for higher-grade inertial sensors. However, existing calibration techniques that target MEMS sensors by avoiding the need for expensive equipment generally do not yet address any of these factors. None of the existing magnitude-based calibration methods include any factors beyond bias, a linear scale factor, and axis misalignment. The geometry-aided technique presented in [73] does include linear specific-force sensitivity in the gyro model and discusses the possibility of using a nonlinear scale factor for the gyros in an appendix; however, its implementation requires a precisely leveled surface. Aside from techniques requiring three-axis positioning devices, one other example of including specific-force sensitivity in the gyro model is found in [63], but its implementation requires a centrifuge.

In theory, magnitude-based calibration methods that use a general-purpose nonlinear parameter estimation technique or the novel in-use calibration method of [78] could be extended to calibrate sensor models with higher-order terms. An obstacle in doing this

would be ensuring that the estimator converges on the correct solution given the added degrees of freedom. Even with relatively simple sensor models, difficulties with convergence and sensitivity to the initial values supplied for the parameters were encountered in [68]. Using nonlinear solution techniques also makes it more difficult to assess how many observations are necessary to obtain robust parameter estimates and to estimate their statistical properties. One other consideration is that a rate table with one or more axes is needed to calibrate gyros using existing magnitude-based techniques, which is a common obstacle in their application to MEMS inertial sensors.

The calibration technique presented here addresses these outstanding limitations. The accelerometer and gyro models presented in Section 3.4 include scale factor nonlinearity, anisotropic sensitivity, and gyro specific-force sensitivity. The calibration method described in Section 3.5 allows these models to be calibrated using standard linear regression. Finally, a method of performing the calibration using simple, inexpensive equipment is demonstrated in Chapter 4.

3.4 Sensor models

The method of orthogonalizing and aligning the accelerometer and gyro axes is presented in Section 3.4.1. Sections 3.4.2 and 3.4.3 then define the accelerometer and gyro models, respectively.

3.4.1 Axis misalignment

When there is no preferred external reference frame, it is common to use the orthogonalized accelerometer axes to define the IMU's reference frame [2, Ch. 2, §III.A]. Orthogonalizing the accelerometer triad requires three misalignment angles. Similarly, three angles are used to orthogonalize the gyro triad and then a final set of three rotation angles is used to align the gyro frame to that of the accelerometers.

An alternate approach adopted here is to directly align the axes of both sensor triads to an external, orthogonal reference frame established by the IMU's case. This method re-

quires two alignment angles per sensor axis giving a total of six angles per triad, as shown in Figure 3.1. In total, three more angles are required than if the orthogonalized accelerometer frame is used as the reference. In return, knowledge of both triads' orientation relative to a fixed reference is obtained and a common approach for calibrating the misalignment can be used for both triads.

Using the angles and axes defined in Figure 3.1, the nonorthogonal transformation from the sensor frame to the case frame, ${}^{c}\mathbf{N}_{s}$, is given by:

$$ {}^{c}\mathbf{N}_{s} = \left[{}^{c}\mathbf{x}_{s} \quad {}^{c}\mathbf{y}_{s} \quad {}^{c}\mathbf{z}_{s} \right] \tag{3.1} $$

where ${}^{c}\mathbf{i}_{s}$ is sensor axis i expressed in the case frame (here and in subsequent equations, the

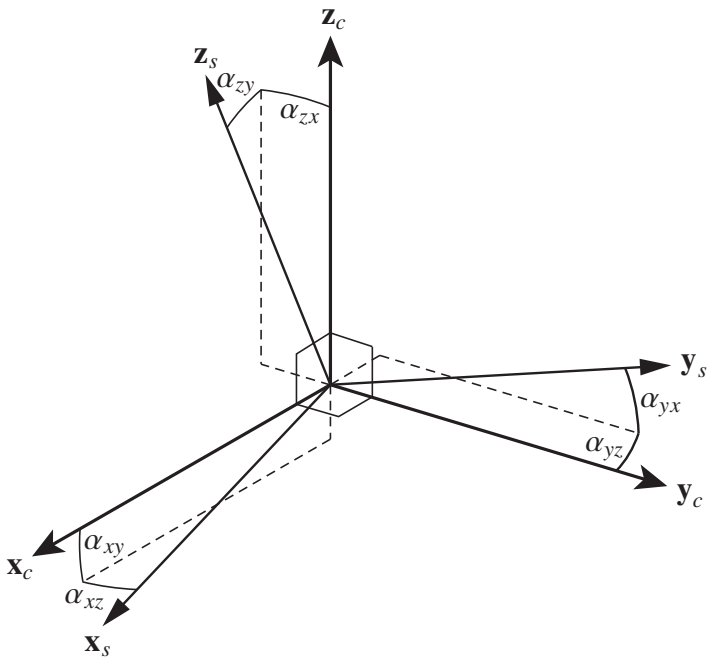


Figure 3.1: Misalignment angles needed to align a set of sensor axes ($\mathbf{x}_s$, $\mathbf{y}_s$, $\mathbf{z}_s$) to an orthogonal reference frame provided by the case ($\mathbf{x}_c$, $\mathbf{y}_c$, $\mathbf{z}_c$). α_{ij} is the rotation angle about case axis j needed to align sensor axis i with the case axis.

left superscript denotes the frame in which a vector is expressed). These can be found via:

$$ {}^{c}\mathbf{x}_s = \mathbf{R}_z\left(\alpha_{xz}\right)\mathbf{R}_y\left(\alpha_{xy}\right){}^{s}\mathbf{x}_s = \begin{bmatrix} \cos\alpha_{xy}\cos\alpha_{xz} \\ \cos\alpha_{xy}\sin\alpha_{xz} \\ -\sin\alpha_{xy} \end{bmatrix}, \tag{3.2a} $$

$$ {}^{c}\mathbf{y}_s = \mathbf{R}_x\left(\alpha_{yx}\right)\mathbf{R}_z\left(\alpha_{yz}\right){}^{s}\mathbf{y}_s = \begin{bmatrix} -\sin\alpha_{yz} \\ \cos\alpha_{yx}\cos\alpha_{yz} \\ \cos\alpha_{yz}\sin\alpha_{yx} \end{bmatrix}, \text{ and} \tag{3.2b} $$

$$ {}^{c}\mathbf{z}_s = \mathbf{R}_y\left(\alpha_{zy}\right)\mathbf{R}_x\left(\alpha_{zx}\right){}^{s}\mathbf{z}_s = \begin{bmatrix} \cos\alpha_{zx}\sin\alpha_{zy} \\ -\sin\alpha_{zx} \\ \cos\alpha_{zx}\cos\alpha_{zy} \end{bmatrix} \tag{3.2c} $$

where $\mathbf{R}_i(\theta)$ is the rotation matrix corresponding to rotation about axis i by angle θ. Note that it is not necessary to assume the misalignment angles are small.

3.4.2 Accelerometer model

Accounting for anisotropic sensitivities and scale factor nonlinearities using a quadratic model, the accelerometer output voltage can be modeled as [1, 2]:[1]

$$ \mathbf{E}_a({}^{c}\mathbf{f}) = \mathbf{K}_{Ea}{}^{a}\mathbf{N}_c{}^{c}\mathbf{f} + \mathbf{A}_{Ea}{}^{c}\mathbf{f}_{ij} + \mathbf{S}_{Ea}{}^{c}\mathbf{f}_{ii} + \mathbf{b}_{Ea} \tag{3.3a} $$

$$ {}^{c}\mathbf{f}_{ij} = \begin{bmatrix} f_x f_y & f_x f_z & f_y f_z \end{bmatrix}^{\mathrm{T}} \tag{3.3b} $$

$$ {}^{c}\mathbf{f}_{ii} = \begin{bmatrix} f_x^2 & f_y^2 & f_z^2 \end{bmatrix}^{\mathrm{T}} \tag{3.3c} $$

[1] To truly be equalities, each of the sensor models presented in this section would have to include an additional term to account for stochastic noise and other unmodeled effects. Since these are not considered in the calibration, they are omitted for the sake of clarity.

where $\mathbf{E}_a$ is a 3×1 vector giving the output voltages from the accelerometer triad, ${}^c\mathbf{f}$ is the specific force vector in the case frame, $\mathbf{K}_{Ea}$ is a diagonal matrix giving the linear scale factors, ${}^a\mathbf{N}_c = ({}^c\mathbf{N}_a)^{-1}$ is the inverse of the nonorthogonal transformation matrix defined in (3.1), $\mathbf{A}_{Ea}$ is a 3×3 matrix of anisotropic sensitivities, ${}^c\mathbf{f}_{ij}$ gives the products of the specific force vector components, $\mathbf{S}_{Ea}$ is a 3×3 matrix of sensitivities to the squared specific forces, ${}^c\mathbf{f}_{ii}$ is a vector of the specific force components squared, and $\mathbf{b}_{Ea}$ is a 3×1 vector of biases. This provides a model of how the sensor is expected to respond to the applied specific force vector, which is useful for simulations.

Unlike the linear models used in the majority of the methods reviewed in Section 3.3, the anisotropic and squared dependencies do not allow (3.3) to be inverted via algebraic and matrix manipulations. Instead, following [2] it is assumed that an analogous relationship with the roles of $\mathbf{E}_a$ and ${}^c\mathbf{f}$ reversed is adequate to allow the applied specific force to be predicted using the output voltages from the accelerometer triad. Given this assumption, the following model is defined:

$$ {}^c\mathbf{f}\left(\mathbf{E}_a\right) = \mathbf{K}_a{}^c\mathbf{N}_a\mathbf{E}_a + \mathbf{A}_a\mathbf{E}_{a,ij} + \mathbf{S}_a\mathbf{E}_{a,ii} + \mathbf{b}_a \, . \tag{3.4} $$

$\mathbf{E}_{a,ij}$, $\mathbf{E}_{a,ii}$, and the coefficient matrices, are all defined analogously to the symbols used in (3.3) with appropriate units to obtain an output dimensioned as an acceleration.

To aid in the calibration, the product of the scale factor and alignment matrices is collapsed as:

$$ {}^c\mathbf{T}_a = \mathbf{K}_a{}^c\mathbf{N}_a = \begin{bmatrix} k_{a,x}{}^c\mathbf{x}_a & k_{a,y}{}^c\mathbf{y}_a & k_{a,z}{}^c\mathbf{z}_a \end{bmatrix} . \tag{3.5} $$

This 3×3 matrix has nine elements from which the three scale factors and six misalignment angles can be extracted using the definitions in (3.2) and the fact that the sensor frame axes are all unit vectors. Together with the other coefficient matrices and bias vector in (3.4),

this gives a total of 30 parameters per triad, 10 per accelerometer axis.

3.4.3 Gyro model

The following model can be used to account for all of the major influences on the gyro outputs described in [1, 2]:

$$\mathbf{E}_g\left({}^c\boldsymbol{\omega}, {}^c\mathbf{f}\right) = \mathbf{K}_{Eg}{}^g\mathbf{N}_c{}^c\boldsymbol{\omega} + \mathbf{A}_{Eg}{}^c\boldsymbol{\omega}_{ij} + \mathbf{S}_{Eg}{}^c\boldsymbol{\omega}_{ii} + \mathbf{F}_{Eg}{}^c\mathbf{f} + \mathbf{A}_{Eg,f}{}^c\mathbf{f}_{ij} + \mathbf{S}_{Eg,f}{}^c\mathbf{f}_{ii} + \mathbf{b}_{Eg}\ . \quad (3.6)$$

Here, $\mathbf{E}_g$ is a 3×1 vector of gyro output voltages, $\mathbf{F}_{Eg}$ is a 3×3 matrix of linear specific-force sensitivities, ${}^c\boldsymbol{\omega}$ is the angular rate in the case frame, and the remaining terms are analogous to those in (3.3). The model accounts for bias, misalignment, scale factor nonlinearity, nonlinear specific-force sensitivity, and anisotropic sensitivity to angular rate and specific force.

As with the model of the accelerometer voltage response, this set of three coupled, nonlinear equations cannot be inverted to solve for the angular rate. Therefore, a similar approach is taken and it is assumed that a sufficient model for the inverse is:

$${}^c\boldsymbol{\omega}\left(\mathbf{E}_g, {}^c\mathbf{f}\right) = \mathbf{K}_g{}^g\mathbf{N}_c\mathbf{E}_g + \mathbf{A}_g\mathbf{E}_{g,ij} + \mathbf{S}_g\mathbf{E}_{g,ii} + \mathbf{F}_g{}^c\mathbf{f} + \mathbf{A}_{g,f}{}^c\mathbf{f}_{ij} + \mathbf{S}_{g,f}{}^c\mathbf{f}_{ii} + \mathbf{b}_g\ . \quad (3.7)$$

Using the same approach and motivation as in (3.5), the product of the scale factor and alignment matrix is collapsed as:

$${}^c\mathbf{T}_g = \mathbf{K}_g{}^g\mathbf{N}_c = \left[k_{g,1}{}^c\mathbf{x}_g \quad k_{g,2}{}^c\mathbf{y}_g \quad k_{g,3}{}^c\mathbf{z}_g\right]. \quad (3.8)$$

Given these definitions, the model in (3.7) includes 57 unknown parameters per triad, 19 per axis.

3.5 Calibration technique

The calibration method is based on using simple geometric constraints much like the techniques described in Section 3.3.3. It is assumed that the equipment used allows the IMU to be placed in known orientations relative to a reference frame. The orientation of the reference frame relative to the local gravity vector need not be known. While in the different orientations, it is assumed that the IMU can be subjected to pure linear motions with known displacements, and to rotations of known angles about an axis whose direction, but not necessarily location, is fixed. Data from these motions and static observations is processed and aggregated to form the regressor matrix and output vector used to calibrate the sensors via least-squares regression.

It is shown in Chapter 4 that the necessary constraints can be achieved using: (1) a rectangular prism as the case for the IMU, (2) support blocks that allow it to be placed in known orientations, (3) a plate with a guide rail and perpendicular reference rails, and (4) a caliper to measure the linear displacements. A diagram showing an example of hardware that can be used to implement the technique is shown in Figure 3.2.

The coordinate frame formed by the $\mathbf{x}_p$, $\mathbf{y}_p$, and $\mathbf{z}_p$ axes in Figure 3.2 is a reference

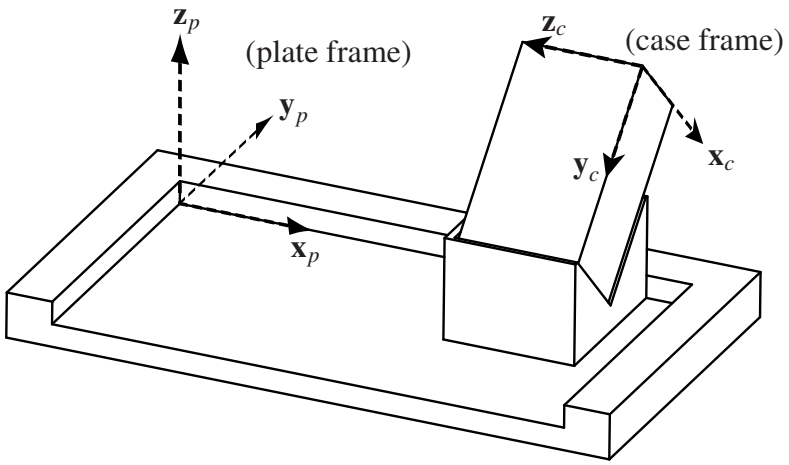


Figure 3.2: Example hardware and coordinate frames used in the calibration. The IMU is shown supported in a V-groove block on the calibration plate. The rear wall provides a guide for executing linear motions and the side walls facilitate measuring the displacements.

frame aligned to the calibration plate's main surface and guide wall. This will be referred to as the plate frame. Knowledge of the geometry of the IMU's case and the support block, when used, allows the rotation matrix relating the plate frame and the frame fixed to the IMU's case, the case frame, to be determined. These assumptions and definitions will be used to present the calibration procedure in this section.

3.5.1 Accelerometer calibration equations

Expressed in the case frame (see Figure 3.2), the equation relating the accelerometer output, the gravity vector, and the kinematic acceleration is:

$$ {}^{c}\mathbf{R}_{p}\,{}^{p}\mathbf{g} + {}^{c}\mathbf{f}\left(\mathbf{E}_{a}(t)\right) = {}^{c}\mathbf{R}_{p}\,{}^{p}\ddot{\mathbf{r}}_{c}(t) \tag{3.9} $$

where $\ddot{\mathbf{r}}_c$ is the second time derivative of the position of the case frame, $\mathbf{g}$ is the gravity vector, $\mathbf{f}\left(\mathbf{E}_a\right)$ is the specific force read by the accelerometers, as determined by the model in (3.4), and ${}^{c}\mathbf{R}_{p}$ is the rotation matrix from the plate frame to the case frame.

Note that in (3.9), $\ddot{\mathbf{r}}_c$ and $\mathbf{g}$ are expressed in the plate frame and then rotated to the case frame. For $\ddot{\mathbf{r}}_c$, this is because the displacements for the linear motions are simple to express in the plate frame. For $\mathbf{g}$, since the calibration plate is stationary, the components of ${}^{p}\mathbf{g}$ are constant and can be included in the list of parameters to be calibrated. This avoids the need to carefully level the calibration plate and is further discussed in Section 3.7.2. Also note that for simplicity, the time dependency of $\ddot{\mathbf{r}}_c$, $\mathbf{E}_a$, and other quantities will generally not be shown explicitly.

3.5.1.1 Static observations

Static observations can be taken by securing the IMU in a known orientation on the calibrate plate either by itself or in a support block. In this case, the kinematic acceleration is zero and (3.9) becomes:

$$ {}^{c}\mathbf{R}_{p}\,{}^{p}\mathbf{g} + {}^{c}\mathbf{f}\left(\mathbf{E}_{a}\right) = \mathbf{0}_{3\times 1} \; . \tag{3.10} $$

Substituting the accelerometer model from (3.4) and (3.5) gives:

$$ {}^{c}\mathbf{R}_{p}\,{}^{p}\mathbf{g} + {}^{c}\mathbf{T}_{a}\mathbf{E}_{a} + \mathbf{A}_{a}\mathbf{E}_{a,ij} + \mathbf{S}_{a}\mathbf{E}_{a,ii} + \mathbf{b}_{a} = \mathbf{0}_{3\times 1} \; . \tag{3.11} $$

A 33×1 vector of the unknown parameters is defined as:

$$ \boldsymbol{\phi}_{a} = \left[{}^{p}\mathbf{g}^{\mathrm{T}} \;\; \mathcal{L}_{r}({}^{c}\mathbf{T}_{a}) \;\; \mathcal{L}_{r}(\mathbf{A}_{a}) \;\; \mathcal{L}_{r}(\mathbf{S}_{a}) \;\; \mathbf{b}_{a}^{\mathrm{T}} \right]^{\mathrm{T}} . \tag{3.12} $$

The operator $\mathcal{L}_{r}(\cdot)$ linearizes a 3×3 matrix by rows as:

$$ \mathcal{L}_{r}(\mathbf{M}_{3\times 3}) = \left[\mathbf{M}_{\mathrm{r}1} \;\; \mathbf{M}_{\mathrm{r}2} \;\; \mathbf{M}_{\mathrm{r}3} \right] \tag{3.13} $$

where $\mathbf{M}_{\mathrm{r}i}$ is the i^{th} row of $\mathbf{M}$. An operator, $\mathcal{D}_{3}(\cdot)$, that operates on 3×1 vectors is also defined to create three transposed copies in the following diagonal format:

$$ \mathcal{D}_{3}(\mathbf{v}_{3\times 1}) = \begin{bmatrix} \mathbf{v}^{\mathrm{T}} & \mathbf{0}_{1\times 3} & \mathbf{0}_{1\times 3} \\ \mathbf{0}_{1\times 3} & \mathbf{v}^{\mathrm{T}} & \mathbf{0}_{1\times 3} \\ \mathbf{0}_{1\times 3} & \mathbf{0}_{1\times 3} & \mathbf{v}^{\mathrm{T}} \end{bmatrix} . \tag{3.14} $$

Using (3.12) and (3.14), (3.11) can be expressed as:

$$ \mathbf{M}_{a,stc}\,\boldsymbol{\phi}_{a} = \mathbf{0}_{3\times 1} \tag{3.15a} $$

$$ \mathbf{M}_{a,stc} = \left[{}^{c}\mathbf{R}_{p} \;\; \mathcal{D}_{3}\left(\overline{\mathbf{E}}_{a}\right) \;\; \mathcal{D}_{3}\left(\overline{\mathbf{E}}_{a,ij}\right) \;\; \mathcal{D}_{3}\left(\overline{\mathbf{E}}_{a,ii}\right) \;\; \mathbf{I}_{3\times 3} \right] . \tag{3.15b} $$

Here, $\mathbf{M}_{a,stc}$ is a 3×30 measurement matrix, $\overline{x}$ denotes the average value of x, $\mathbf{I}_{3\times3}$ is a 3×3 identity matrix, and the output vector is a 3×1 zero vector. This form allows the voltage readings to be sampled over a finite period while the IMU is stationary in the desired orientation and then averaged to minimize the effects of output noise for that observation.

3.5.1.2 Linear motion observations

Next, consider linear displacements where the IMU and support block, if used, are translated a known distance without rotation such that ${}^{c}\mathbf{R}_{p}$ is constant. The kinematic acceleration is no longer zero, but the geometrical constraints ensure that it is directly purely along the $\mathbf{x}_p$ axis for the plate frame definition shown in Figure 3.2. Therefore, the displacement can be measured and the velocity at the beginning and end of the motion are both known to be zero.

Defining the case frame velocity as $\mathbf{v}_c = \dot{\mathbf{r}}_c$ and integrating the right-hand side of (3.9) once with respect to (w.r.t.) time over the duration of the motion gives:

$$\int_{t_0}^{t_f} {}^{c}\mathbf{R}_{p} \frac{d\mathbf{v}_c}{dt}\, dt = {}^{c}\mathbf{R}_{p} \int_{\mathbf{v}_{c,0}}^{\mathbf{v}_{c,f}} d\mathbf{v}_c = {}^{c}\mathbf{R}_{p} \left(\mathbf{v}_{c,f} - \mathbf{v}_{c,0}\right) = \mathbf{0}_{3\times1} \tag{3.16}$$

where t_0 and t_f are the time at the beginning and end of the motion, respectively, $\mathbf{v}_{c,0}$ is the initial velocity of the case frame, and $\mathbf{v}_{c,f}$ is its final velocity. The velocity vectors, $\mathbf{v}_c$, $\mathbf{v}_{c,0}$, and $\mathbf{v}_{c,f}$ are understood to be expressed in the plate frame although this designation is omitted for clarity. As noted above, $\mathbf{v}_{c,0} = \mathbf{v}_{c,f} = \mathbf{0}_{3\times1}$ since the IMU starts and stops at rest.

Integrating the left-hand side of (3.9) w.r.t. time gives:

$$\int_{t_0}^{t_f} \left[{}^{c}\mathbf{R}_{p}\, {}^{p}\mathbf{g} + {}^{c}\mathbf{f}\left(\mathbf{E}_a\right)\right] dt = {}^{c}\mathbf{R}_{p}\, {}^{p}\mathbf{g} \left(t_f - t_0\right) + {}^{c}\mathbf{T}_{a} \int_{t_0}^{t_f} \mathbf{E}_a\, dt + \cdots$$

$$\cdots\ \mathbf{A}_a \int_{t_0}^{t_f} \mathbf{E}_{a,ij}\, dt + \mathbf{S}_a \int_{t_0}^{t_f} \mathbf{E}_{a,ii}\, dt + \mathbf{b}_a \left(t_f - t_0\right) . \tag{3.17}$$

Since the integrals that remain in (3.17) are definite integrals, the mean value theorem can be applied. Recall that it states:

$$\int_a^b h(t)\,dt = \overline{h}(t)\Big|_a^b \tag{3.18}$$

where the right hand side denotes the average value of $h(t)$ over the interval $[a, b]$. Applying this in (3.17) gives:

$${}^c\mathbf{R}_p\,{}^p\mathbf{g}\,\Delta t + {}^c\mathbf{T}_a\overline{\mathbf{E}}_a\Delta t + \mathbf{A}_a\overline{\mathbf{E}}_{a,ij}\Delta t + \mathbf{S}_a\overline{\mathbf{E}}_{a,ii}\Delta t + \mathbf{b}_a\Delta t \tag{3.19a}$$

$$\Delta t = t_f - t_0\ . \tag{3.19b}$$

Remembering that this expression is equal to (3.16), a zero vector, the Δt factors can all be divided out. Doing this, recombining the two sides of the equation, and rearranging to group the unknown parameters into the vector $\boldsymbol{\phi}_a$ results in the same expression as was found for the static observations (3.15). Hence, the three calibration equations represented in (3.15a) apply for both static and linear motions.

Next, return to (3.9) and integrate first to an indefinite time, t, so that a second integration can then be performed to obtain an equation in terms of position. Beginning with the right-hand side of (3.9), the first integral yields ($\mathbf{v}_c$ is again to be expressed in the plate frame):

$$\int_{t_0}^{t} {}^c\mathbf{R}_p\frac{d\mathbf{v}_c}{dt}\,dt = {}^c\mathbf{R}_p\int_{\mathbf{v}_{c,0}}^{\mathbf{v}_c(t)} d\mathbf{v}_c' = {}^c\mathbf{R}_p\mathbf{v}_c(t)\ . \tag{3.20}$$

Integrating again from t_0 to t_f gives:

$$\int_{t_0}^{t_f} {}^{c}\mathbf{R}_p \frac{d\mathbf{r}_c}{dt}\, dt = {}^{c}\mathbf{R}_p \int_{\mathbf{r}_{c,0}}^{\mathbf{r}_{c,f}} d\mathbf{r}_c = {}^{c}\mathbf{R}_p {}^{p}\Delta\mathbf{r}_c \tag{3.21a}$$

$${}^{p}\Delta\mathbf{r}_c = {}^{p}\mathbf{r}_{c,f} - {}^{p}\mathbf{r}_{c,0} \tag{3.21b}$$

where it is emphasized that the displacement vector, $\Delta\mathbf{r}_c$, is expressed in the plate frame where it is simple to measure. For the setup in Figure 3.2, its form would be:

$${}^{p}\Delta\mathbf{r}_c = \begin{bmatrix} d & 0 & 0 \end{bmatrix}^{\mathrm{T}} \tag{3.22}$$

with d being the signed displacement.

Integrating the left-hand side of (3.9) first from t_0 to t and then from t_0 to t_f and applying the mean value theorem (3.18) for the second integration results in the following expression:

$${}^{c}\mathbf{R}_p {}^{p}\mathbf{g}\frac{\Delta t^2}{2} + {}^{c}\mathbf{T}_a \overline{\int_t \mathbf{E}}_a \Delta t + \mathbf{A}_a \overline{\int_t \mathbf{E}}_{a,ij} \Delta t + \mathbf{S}_a \overline{\int_t \mathbf{E}}_{a,ii} \Delta t + \mathbf{b}_a \frac{\Delta t^2}{2} \tag{3.23}$$

where the shorthand notation below is employed.

$$\overline{\int_t h} = \underset{t=[t_0,t_f]}{\mathrm{mean}} \left(\int_{t_0}^{t} h(\tau)\, d\tau \right) \tag{3.24}$$

The result in (3.23) is equal to the displacement vector rotated into the case frame in (3.21a). This provides another set of three equations to use in the calibration, but they have units of length rather than acceleration. To remedy this and provide proper weighting for the errors when the equations are included in the regression, both sides are divided by $\Delta t^2/2$. This scales the values to be equivalent to the constant acceleration that would result in the same amount of displacement in the given time period. The error magnitudes will

then be directly comparable to those of the static observation equation in (3.15a). Equating (3.21a) and (3.23), applying this scaling, and rearranging to group the unknowns into the vector $\boldsymbol{\phi}_a$ gives:

$$\mathbf{M}_{a,lin}\,\boldsymbol{\phi}_a = \mathbf{u}_{a,lin} \tag{3.25a}$$

$$\mathbf{M}_{a,lin} = \left[{}^{c}\mathbf{R}_p \;\; \mathcal{D}_3\left(\tfrac{2}{\Delta t}\overline{\textstyle\int_t \mathbf{E}}_a\right) \;\; \mathcal{D}_3\left(\tfrac{2}{\Delta t}\overline{\textstyle\int_t \mathbf{E}}_{a,ij}\right) \;\; \mathcal{D}_3\left(\tfrac{2}{\Delta t}\overline{\textstyle\int_t \mathbf{E}}_{a,ii}\right) \;\; \mathbf{I}_{3\times 3}\right] \tag{3.25b}$$

$$\mathbf{u}_{a,lin} = \frac{2}{\Delta t^2}{}^{c}\mathbf{R}_p\,{}^{p}\Delta\mathbf{r}_c \tag{3.25c}$$

where $\mathbf{M}_{a,lin}$ is the 3×33 measurement matrix and $\mathbf{u}_{a,lin}$ is the 3×1 output vector.

3.5.2 Gyro calibration equations

For the gyro calibration, it is assumed that the accelerometer triad has already been calibrated so that it can be used to provide the specific force in the case frame via (3.4). Formally, the angular rate sensed by the gyros during calibration is:

$${}^{c}\boldsymbol{\omega}\left(\mathbf{E}_g, {}^{c}\mathbf{f}\right) = {}^{c}\mathbf{R}_p\left({}^{p}\boldsymbol{\omega}_{c/p} + {}^{p}\boldsymbol{\omega}_{E/i}\right) \tag{3.26}$$

where ${}^{p}\boldsymbol{\omega}_{c/p}$ is the angular velocity of the IMU case frame relative to the plate frame and ${}^{p}\boldsymbol{\omega}_{E/i}$ is the angular velocity of the Earth relative to an inertial frame. As discussed in Section 3.7.3, MEMS gyros cannot accurately detect Earth's rotation rate so it will be neglected in the initial development of the calibration equations presented in Sections 3.5.2.1 and 3.5.2.2. However, Section 3.5.2.3 demonstrates modifications that allow it to be included by calibrating its components individually analogously to the treatment of ${}^{p}\mathbf{g}$ in the accelerometer calibration. While the magnitude of $\boldsymbol{\omega}_{E/i}$ is constant and independent of geographical location, the motivations for separately calibrating its components, as

discussed in Section 3.7.2 regarding ${}^{p}\mathbf{g}$, still apply.

3.5.2.1 Static and linear motion observations

Neglecting Earth's rotation rate, (3.26) reduces to:

$$ {}^{c}\boldsymbol{\omega}\left(\mathbf{E}_g, {}^{c}\mathbf{f}\right) = \mathbf{0}_{3\times 1} \tag{3.27} $$

for static and linear motion observations since ${}^{p}\boldsymbol{\omega}_{c/p} = \mathbf{0}_{3\times 1}$ during both. Using the linearizing operator defined in (3.13), the unknown coefficients from the gyro model given in (3.7) and (3.8) are grouped into a single 57×1 vector defined as:

$$ \boldsymbol{\phi}_g = \left[\mathcal{L}_r\left({}^{c}\mathbf{T}_g\right) \;\; \mathcal{L}_r\left(\mathbf{A}_g\right) \;\; \mathcal{L}_r\left(\mathbf{S}_g\right) \;\; \mathcal{L}_r\left(\mathbf{F}_g\right) \;\; \mathcal{L}_r\left(\mathbf{A}_{g,f}\right) \;\; \mathcal{L}_r\left(\mathbf{S}_{g,f}\right) \;\; \mathbf{b}_g^{\mathrm{T}}\right]^{\mathrm{T}}. \tag{3.28} $$

Using this definition and the $\mathcal{D}_3(\cdot)$ operator defined in (3.14), (3.27) can be expressed as:

$$ \mathbf{M}_{g,stc}\,\boldsymbol{\phi}_g = \mathbf{0}_{3\times 1} \tag{3.29a} $$

$$ \mathbf{M}_g = \left[\mathcal{D}_3\left(\overline{\mathbf{E}}_g\right) \;\; \mathcal{D}_3\left(\overline{\mathbf{E}}_{g,ij}\right) \;\; \mathcal{D}_3\left(\overline{\mathbf{E}}_{g,ii}\right) \;\; \mathcal{D}_3\left({}^{c}\overline{\mathbf{f}}\right) \;\; \mathcal{D}_3\left({}^{c}\overline{\mathbf{f}}_{ij}\right) \;\; \mathcal{D}_3\left({}^{c}\overline{\mathbf{f}}_{ii}\right) \;\; \mathbf{I}_{3\times 3}\right] \tag{3.29b} $$

where $\mathbf{M}_g$ is a 3×57 measurement matrix and the output vector is a 3×1 zero vector.

3.5.2.2 Rotation observations

Next, consider rotations that are performed while keeping the IMU and support block, if used, in contact with the main surface of the calibration plate. Using the geometry of these components and the walls of the plate as references, rotations of known angles can be performed by starting and stopping with a given surface of the IMU or support block against one of the plate walls. Since the rotation is about a fixed axis, the direction of the

angular velocity vector ${}^{p}\boldsymbol{\omega}_{c/p}$ is fixed. For the setup in Figure 3.2, it is simply:

$$ {}^{p}\boldsymbol{\omega}_{c/p}(t) = \begin{bmatrix} 0 & 0 & {}^{p}\omega_{z}(t) \end{bmatrix}^{\mathrm{T}} . \tag{3.30} $$

Integrating w.r.t. time gives:

$$ \int_{t_0}^{t_f} {}^{p}\boldsymbol{\omega}_{c/p}(t)\, dt = \begin{bmatrix} 0 & 0 & \theta_{z_p} \end{bmatrix}^{\mathrm{T}} \tag{3.31} $$

where θ_{z_p} is the known rotation angle of the case frame relative to the plate frame about the $\mathbf{z}_p$ axis and t_0 and t_f are the time at the beginning and end of the rotation, respectively.

During the rotation, the orientation of the case frame relative to the plate frame can be expressed as:

$$ {}^{c}\mathbf{R}_{p}(t) = {}^{c}\mathbf{R}_{c_0}(t)\, {}^{c_0}\mathbf{R}_{p} = \mathbf{R}_{z_p}\left(\theta_{z_p}(t)\right) {}^{c_0}\mathbf{R}_{p} \tag{3.32} $$

where ${}^{c_0}\mathbf{R}_{p}$ is the rotation from the plate frame to the initial orientation of the case frame, and ${}^{c}\mathbf{R}_{c_0}(t) = \mathbf{R}_{z_p}\left(\theta_{z_p}(t)\right)$ is the rotation from the initial to the instantaneous orientation of the case frame, which is a rotation about the $\mathbf{z}_p$ axis. Since $\boldsymbol{\omega}_{c/p}$ is parallel to the $\mathbf{z}_p$ axis, the time-dependent portion of the rotation does not affect its components and the following is true:

$$ {}^{c}\boldsymbol{\omega}_{c/p}(t) = {}^{c_0}\boldsymbol{\omega}_{c/p}(t) = {}^{c_0}\mathbf{R}_{p}\, {}^{p}\boldsymbol{\omega}_{c/p}(t) . \tag{3.33} $$

Neglecting Earth's rotation rate, using (3.33) in (3.26) gives:

$$ {}^{c}\boldsymbol{\omega}\left(\mathbf{E}_{g}, {}^{c}\mathbf{f}\right) = {}^{c_0}\mathbf{R}_{p}\, {}^{p}\boldsymbol{\omega}_{c/p}(t) . \tag{3.34} $$

Noting that ${}^{c_0}\mathbf{R}_p$ is constant, substituting the gyro sensor model from (3.7) and (3.8), integrating both sides from t_0 to t_f, utilizing (3.31), and applying the mean value theorem (3.18) gives:

$$ {}^{c}\mathbf{T}_g\overline{\mathbf{E}}_g\Delta t + \mathbf{A}_g\overline{\mathbf{E}}_{g,ij}\Delta t + \mathbf{S}_g\overline{\mathbf{E}}_{g,ii}\Delta t + \mathbf{F}_g\,{}^{c}\overline{\mathbf{f}}\,\Delta t + \mathbf{A}_{g,f}\,{}^{c}\overline{\mathbf{f}}_{i,j}\Delta t + \cdots $$

$$ \cdots\ \mathbf{S}_{g,f}\,{}^{c}\overline{\mathbf{f}}_{i,i}\Delta t + \mathbf{b}_g\Delta t = {}^{c_0}\mathbf{R}_p\begin{bmatrix}0 & 0 & \theta_{z_p}\end{bmatrix}^{\mathrm{T}} \quad (3.35) $$

where Δt is the rotation time span defined identically to (3.19b). Dividing both sides by Δt makes the right-hand side equal to the average angular rate during the rotation so that the errors will have comparable magnitudes and consistent units with the static observation equation in (3.29). Doing this and reformatting the left-hand side in terms of $\boldsymbol{\phi}_g$ gives:

$$ \mathbf{M}_g\,\boldsymbol{\phi}_g = \mathbf{u}_{g,rot} \quad (3.36a) $$

$$ \mathbf{u}_{g,rot} = {}^{c_0}\mathbf{R}_p\begin{bmatrix}0 & 0 & \frac{\theta_{z_p}}{\Delta t}\end{bmatrix}^{\mathrm{T}} \quad (3.36b) $$

where $\mathbf{M}_g$ is identical to the static observation measurement matrix defined in (3.29b) and $\mathbf{u}_{g,rot}$ is the 3×1 output vector.

3.5.2.3 Modifications to include Earth's rotation rate

For gyros with sensitivities and signal-to-noise ratios that allow Earth's rotation rate to be detected, the static observation equation is obtained by setting $\boldsymbol{\omega}_{c/p}$ to zero in (3.26) yielding:

$$ {}^{c}\boldsymbol{\omega}\left(\mathbf{E}_g, {}^{c}\mathbf{f}\right) - {}^{c}\mathbf{R}_p\,{}^{p}\boldsymbol{\omega}_{E/i} = \mathbf{0}_{3\times 1} \quad (3.37) $$

Augmenting $\boldsymbol{\phi}_g$ to include the three components of ${}^{p}\boldsymbol{\omega}_{E/i}$, this can be written using:

$$\mathbf{M}_{g,\omega_E,stc}\,\boldsymbol{\phi}_{\omega_E} = \mathbf{0}_{3\times 1} \tag{3.38a}$$

$$\mathbf{M}_{g,\omega_E,stc} = \begin{bmatrix}\mathbf{M}_g & -{}^{c}\mathbf{R}_p\end{bmatrix} \tag{3.38b}$$

$$\boldsymbol{\phi}_{g,\omega_E} = \begin{bmatrix}\boldsymbol{\phi}_g^{\mathrm{T}} & {}^{p}\boldsymbol{\omega}_{E/i}{}^{\mathrm{T}}\end{bmatrix}^{\mathrm{T}} \tag{3.38c}$$

where $\mathbf{M}_g$ and $\boldsymbol{\phi}_g$ are as defined in (3.29b) and (3.28), respectively.

The finite rotations are complicated by the fact that without careful alignment, $\boldsymbol{\omega}_{E/i}$ will not be parallel with the rotation axis, $\mathbf{z}_p$. Using (3.33) in (3.26) gives:

$${}^{c}\boldsymbol{\omega}\left(\mathbf{E}_g, {}^{c}\mathbf{f}\right) - {}^{c}\mathbf{R}_p(t)\,{}^{p}\boldsymbol{\omega}_{E/i} = {}^{c_0}\mathbf{R}_p\,{}^{p}\boldsymbol{\omega}_{c/p}(t)\,. \tag{3.39}$$

Integrating w.r.t time from t_0 to t_f, applying the mean value theorem (3.18), and dividing by Δt yields:

$$\mathbf{M}_{g,\omega_E,rot}\,\boldsymbol{\phi}_{\omega_E} = \mathbf{u}_{g,rot} \tag{3.40a}$$

$$\mathbf{M}_{g,\omega_E,rot} = \begin{bmatrix}\mathbf{M}_g & -\overline{{}^{c}\mathbf{R}_p(t)}\Big|_{t_0}^{t_f}\end{bmatrix} \tag{3.40b}$$

where $\mathbf{u}_{g,rot}$ is as defined in (3.36b) and the dependency of ${}^{c}\mathbf{R}_p$ on time and the interval over which its element-wise average should be evaluated are shown explicitly to clarify the calculation method. The result is not a valid rotation matrix, in general, nor does it represent the orientation of $\boldsymbol{\omega}_{E/i}$. Rather, it provides the net contribution the components of $\boldsymbol{\omega}_{E/i}$ on each gyro axis over the course of the rotation.

Evaluating (3.40b) requires knowledge of ${}^{c}\mathbf{R}_p(t)$, which can be obtained given $\theta_{z_p}(t)$, the

time history of the rotation angle about $\mathbf{z}_p$. The most accurate way of obtaining this would be to use an external measurement system. In the absence of such equipment, assuming small misalignments between the gyro axes and the case frame, $\theta_{z_p}(t)$ can be estimated from the uncalibrated gyro output using:

$$\theta_{z_p}(t) \approx \frac{\int_t E_{g|z_p}(t)}{\int_t E_{g|z_p}(t_f)}\, \theta_{z_p}(t_f) \tag{3.41a}$$

$$\int_t E_{g|z_p}(t) = \int_{t_0}^{t} \Big(\mathbf{E}_g(\tau) - \mathbf{E}_g(t_0)\Big) \cdot {}^{c_0}\mathbf{z}_p \, d\tau \; . \tag{3.41b}$$

This approach relies on the fact that, neglecting nonlinearities, the gyro outputs are shifted and scaled versions of the angular rate. Since the rotation axis is fixed, the projection of $\boldsymbol{\omega}_{E/i}$ and $\mathbf{g}$ on that axis and their contributions to the gyro output will be constant during the rotation. Hence, the projection of the shifted gyro output voltage on the rotation axis in (3.41b) varies only due to $\boldsymbol{\omega}_{c/p}$ and the rotation angle can be calculated using the scaling provided by (3.41a). Error in the estimate of $\theta_{z_p}(t)$ will result only from nonlinearities, which are typically on the order of 1% or less even for consumer-grade MEMS gyros.

3.5.3 Estimating the parameters

For both types of sensors, the parameters are estimated using standard linear regression. Given m observations, the measurement and output matrices from each observation are stacked into the regressor matrix, $\widetilde{\mathbf{M}}$, and composite output vector, $\tilde{\mathbf{u}}$, to give the following set of equations:

$$\widetilde{\mathbf{M}}\,\boldsymbol{\phi} = \tilde{\mathbf{u}} \tag{3.42a}$$

$$\text{where: } \widetilde{\mathbf{M}} = \begin{bmatrix} \mathbf{M}^{[1]} \\ \vdots \\ \mathbf{M}^{[m]} \end{bmatrix} \text{ and: } \tilde{\mathbf{u}} = \begin{bmatrix} \mathbf{u}^{[1]} \\ \vdots \\ \mathbf{u}^{[m]} \end{bmatrix}. \tag{3.42b}$$

In (3.42), $\boldsymbol{\phi}$ is either $\boldsymbol{\phi}_a$ or $\boldsymbol{\phi}_g$, $\mathbf{M}^{[i]}$ is the applicable measurement matrix for observation i, and $\mathbf{u}^{[i]}$ is the corresponding output vector. Enough linearly independent observations must be taken that the rank of $\widetilde{\mathbf{M}}$ equals the number of parameters to be estimated. Then, given this overdetermined system of equations, the estimate of the parameter vector, $\hat{\boldsymbol{\phi}}$, is obtained via the standard least-squares solution:

$$\hat{\boldsymbol{\phi}} = \left(\widetilde{\mathbf{M}}^{\mathrm{T}} \widetilde{\mathbf{M}}\right)^{-1} \widetilde{\mathbf{M}}^{\mathrm{T}} \tilde{\mathbf{u}} \tag{3.43}$$

No weighting matrix is needed because the measurement matrices and output vectors are defined with proper scaling to ensure consistency of units and error magnitudes.

For the accelerometers, (3.15) applies to both static and linear observations and (3.25) also applies for linear observations. Hence, each static observation contributes three rows to the stacked regressor matrix and each linear motion contributes six. Since there are 33 parameters to be estimated, enough static and linear observations must be used to give at least this many linearly independent rows in the regressor.

For the gyros, (3.29) applies to both static and linear observations and (3.36) applies to rotation observations. Both contribute three rows to the regressor matrix. With a total of 57 parameters, this means a minimum of 19 observations is necessary.

For both types of sensors, static observations cannot be used exclusively or the resulting output vector will have no nonzero elements to set the scale of the parameters. While the minimum number of observations necessary to obtain a solution has been defined in this section, the resulting parameter estimates would be very sensitive to errors in the data. Selecting a sufficiently large and diverse set of observations to ensure robust parameter

estimates is discussed in detail in Chapter 4.

3.6 Evaluating the calibration results

3.6.1 Regressor condition number

Singular value decomposition (SVD) is an effective method of evaluating the robustness of the parameter estimates given the regressor matrix. The parameter estimates from (3.43) are effectively divided by the singular values of the regressor matrix [79, §6.3]. Hence, if any of the singular values are very small, this indicates that one or more of the parameter estimates will have high sensitivity to errors in the input data. This sensitivity can degrade the overall accuracy of the calibration results.

Since the parameters of the sensor models can have widely varying magnitudes, the columns of the regressor matrix should be scaled before comparing the singular values [80, Ch. 25]. An effective method is to scale each column by its norm using:

$$\widetilde{\mathbf{M}}' = \widetilde{\mathbf{M}}\mathbf{H} \tag{3.44a}$$

$$\mathbf{H} = \operatorname{diag}(h_1 , h_2 , \dots , h_n) \tag{3.44b}$$

$$h_i = \begin{cases} \left\| \widetilde{\mathbf{M}}_{ci} \right\|^{-1} & \text{if } \left\| \widetilde{\mathbf{M}}_{ci} \right\| \neq 0 \\ 1 & \text{otherwise} \end{cases} \tag{3.44c}$$

where $\widetilde{\mathbf{M}}_{ci}$ is column i of $\widetilde{\mathbf{M}}$, n is the number of columns, and $\mathbf{H}$ is a diagonal matrix constructed with the indicated values.

After applying this scaling, the singular values of $\widetilde{\mathbf{M}}'$ can be used to determine the condition number, c, via:

$$c = \frac{\mu_1}{\mu_n} \tag{3.45}$$

where μ_1 is the largest singular value and μ_n is the smallest. A small condition number indicates that the observability of the parameters is balanced and they will have similar sensitivities to errors in the data. A large condition number indicates that the observations do not provide enough information to estimate all of the parameters robustly. In this case, either additional observations should be included to improve the observability of the parameters, or it may be necessary to remove parameters with poor observability from the model. Hollerbach and Wampler [81] indicate that such measures should be pursued if the condition number is greater than 100. This will be further discussed regarding simulated and experimental results in Chapter 4.

3.6.2 Residual error

The residual errors from the aggregated calibration equations (3.42) can also be used to assess the accuracy of the calibration. Due to noise and measurement error, the relationship in (3.42) is not a true equality. After obtaining the parameter estimates via (3.43), the residual errors in the observation equations are:

$$\mathbf{e} = \tilde{\mathbf{u}} - \widetilde{\mathbf{M}}\hat{\boldsymbol{\phi}} \,. \tag{3.46}$$

The elements of $\mathbf{e}$ are expected to be normally distributed with zero-mean. If this is not true, there may be a systematic error in the method of data collection. Outliers in $\mathbf{e}$ can be used to help identify and remove observations that may have been performed incorrectly or inaccurately.

Since the observation equations are formulated to provide meaningful error magnitudes with consistent units of acceleration and angular rate, the residual errors can also be used to provide an estimate of the variance in the calibrated sensor output using [82, §5.3.3]:

$$\nu^2 = \frac{\mathbf{e}^{\mathrm{T}}\mathbf{e}}{n_r - n_c} \tag{3.47}$$

where n_r and n_c are the number of rows and columns in $\widetilde{\mathbf{M}}$, respectively. Using the formulation presented in Section 3.5, this value will apply to each of the three axes of the sensor triads since they are calibrated simultaneously.

3.6.3 Parameter variance

An estimate of the parameter covariance matrix, $\mathbf{C}_\phi$, can be obtained using:

$$\mathbf{C}_\phi = \nu^2 \left(\widetilde{\mathbf{M}}^{\mathrm{T}} \widetilde{\mathbf{M}} \right)^{-1} . \tag{3.48}$$

Details regarding the derivation of this relationship can be found in [82, §5.3.3]. The elements on the diagonal of $\mathbf{C}_\phi$ are the variances of the individual parameter estimates and reflect how much uncertainty there is in their calibrated values. The estimated standard deviation for parameter i is:

$$\sigma_i = \sqrt{\mathbf{C}_{\phi,(i,i)}} \tag{3.49}$$

where $\mathbf{C}_{\phi,(i,i)}$ is element (i, i) of $\mathbf{C}_\phi$.

3.7 Discussion

3.7.1 Integration of the calibration equations

As demonstrated in Section 3.5, the formulation of the observation equations for linear motions and rotations eliminates the need for precise knowledge of the specific forces and angular velocities applied to the IMU. Integrating the equations with respect to time allows time-integrated and time-averaged sensor outputs to be compared to linear displacements and rotation angles rather than equating raw sensor outputs to specific forces and angular rates. This simplifies and improves the accuracy of the calibration process since displace-

ments and rotation angles can be controlled and measured more easily and with greater precision than their time derivatives.

This approach is similar in spirit to a method presented in [83] wherein time integration was also used to eliminate acceleration from the governing equations for a different calibration problem. In this study, data from a force-torque sensor and joint encoders were used in the Newton-Euler equations of motion to estimate the inertial parameters of a load attached to a robotic manipulator. It was demonstrated that integrating the equations of motion once with respect to time allows linear and angular accelerations to be eliminated from the calibration equations. Otherwise, these accelerations would have to be obtained through double numerical differentiation of joint encoder data. This is undesirable since the resulting acceleration estimates typically have a very poor signal-to-noise ratio.

Difficulties arose in this application since it required integration of the force and torque data. This was found to introduce more error than double differentiation of the encoder data due to bias in the force-torque sensor's outputs. Therefore, the integrated versions of the equations were not used to obtain the results presented in the study. Given this drawback, techniques for inertial parameter estimation have more commonly either:

- Retained the acceleration terms and filtered the numerically differentiated data [84, 85]
- Used energy- or power-based methods to avoid acceleration terms [86–89]
- Used low-pass filtering of the Newton-Euler equations to eliminate the acceleration terms [90–92]

For the calibration technique presented herein, integrating the equations with respect to time does not create concerns like those encountered in [83]. The biases of the inertial sensors are included in the list of parameters to be calibrated and, therefore, do not create unmodeled errors. Also, the reference linear and rotary displacements can be measured directly rather than requiring integration of another sensor's output. Therefore, the integrated equations can be used without compromising the accuracy of the calibration results.

3.7.2 Calibrating gravity as an unknown

Including the components of ${}^{p}\mathbf{g}$ in the list of unknown parameters offers three main benefits: (1) the calibration plate does not need to be leveled precisely, (2) knowledge of the local value of the gravitational acceleration is not required, and (3) the model remains linear. From the three components, the magnitude of the local gravity vector and the two tilt angles of the calibration plate can be derived. Therefore, when none of these values are known, solving for the vector components individually does not increase the number of unknowns.

Regarding the first benefit, current MEMS accelerometers theoretically can resolve tilt angles as small as 0.06° [78]. Ensuring that leveling errors would be negligible relative to the accelerometer sensitivity would, therefore, require a precision-grade level. Such devices are not generally available to researchers seeking to calibrate MEMS accelerometers.

Errors due to imperfect knowledge of the local gravitational acceleration are likely to be small, but are worthy of consideration. As an example, the location where this research was conducted is at an elevation of approximately 1.4 km above sea level. The local gravitational acceleration, as reported by the National Geodetic Survey in June of 1998, is 9.7977 m/s^2, which is 0.09% lower than the standard value of 9.80665 m/s^2. If the standard value were used as the only reference, as would be done in a magnitude-based calibration technique, the calibrated accelerometer outputs would be proportionally in error.

For locations in the United States, interpolated gravity values can be obtained via the NGS Surface Gravity Prediction utility [93]. Even though this can provide values with negligible error, it is still desirable to solve for the three components of the gravity vector individually. Unless the precise tilt angles of the plate are known, eliminating the magnitude of gravity as a variable and retaining the angles would introduce trigonometric functions of unknown parameters requiring the use of a nonlinear parameter estimation technique. Moderately accurate leveling such that small angle assumptions could be used for the tilt angles could circumvent this problem, but estimating the gravity vector during

the calibration provides an excellent metric to gage the accuracy of the results. Even if the exact local gravitational acceleration is not known, it is easy to tell if there is significant error in the calibrated magnitude or direction, indicating a problem in the data collection or processing.

As a final note, if ${}^{p}\mathbf{g}$ is known through careful leveling or the results of previous calibrations on stationary equipment, it can be eliminated from $\boldsymbol{\phi}_a$ and included in the output vector of the observation equations to reduce the number of unknowns. Fewer observations will then be needed to estimate the remaining parameters.

3.7.3 Neglecting Earth's rotation rate

The IMU used to test the calibration method in Chapter 4 will be used as an example to justify neglecting Earth's rotation rate. The most sensitive gyro triad has a dynamic range of 30°/s and nominal sensitivity of 33.3 mV/(°/s). Thus, Earth's rotation rate of approximately 15°/ hr would cause a maximum change of only 0.139 mV in the output of any axis. The resolution of the 16-bit ADC used to sample the output is 0.153 mV and the noise in the gyro's output has an amplitude of approximately 40 mV. Hence, the rotation of the Earth could cause at most a change of 1 division in the ADC reading, but this would be entirely masked by the noise, which is nearly 300 times larger.

3.7.4 Model parameters

The sensor models presented here are, by design, very complex. Depending on the type and accuracy-grade of the sensors used in the IMU, some of the parameters may not apply or may be negligible compared to the noise and instability present in their output. This can be objectively assessed by attempting to calibrate all of the parameters and reviewing their variance and impact on the residual errors. If the standard deviation of a parameter is very high relative to its magnitude, this indicates that there is little confidence in the calibrated value. It can be eliminated by setting the parameter to zero and removing the corresponding

column from the regressor matrix. The residual errors in the calibration equations with and without the parameter can be compared to decide if the parameter should be retained.

The calibration method also provides significant flexibility to explore the inclusion of other terms in the sensor models. Any function of the sensor voltages can be included, such as absolute values or square roots, so long as the parameters to be calibrated are linear multipliers of these functions. If temperature sensors are available, dependency on functions of their outputs could also be included. The drawback associated with these types of modifications is the increase in the number of observations that will be required to identify more parameters.

3.8 Conclusion

The calibration technique presented here overcomes the limitations of existing MEMS IMU calibration techniques discussed in Section 3.3.5. It allows scale factor nonlinearities, anisotropic effects, and nonlinear gyro specific-force sensitivity to be included in the sensor models. The novel mathematical approach used to develop the calibration equations eliminates the need for angular rate measurements and precise alignment relative to the gravity vector. It also allows the model parameters to be estimated using linear regression. This allows standard analysis techniques to be used to estimate the variance of the parameter estimates and assess whether or not a given set of observations will provide numerically robust results. The approach also makes it easy to add and remove parameters in the sensor models to explore different dependencies and to define the values of any parameters that are known before-hand to reduce the number of observations required to estimate the others.

The method can be implemented using simple fixtures and manual excitation of the sensors thereby avoiding the need for costly equipment. Aside from knowledge of the geometry of the fixtures, the only other measurements necessary are distance measurements that can be obtained using a caliper. Simulated and experimental results demonstrating the successful application of the calibration method are presented in Chapter 4.

CHAPTER 4

A POWERFUL AND INEXPENSIVE METHOD FOR CALIBRATING AN INERTIAL MEASUREMENT UNIT: IMPLEMENTATION

Eric Allen Johnson and Stacy J. Morris Bamberg

Department of Mechanical Engineering, University of Utah

Salt Lake City, UT 84112

The contents of this chapter have been submitted to *Sensors and Actuators A: Physical* for review. Formatting and other minor modifications have been applied for consistency with the remainder of this dissertation.

4.1 Abstract

This chapter describes the implementation of a new calibration technique for inertial sensors. The technique is well-suited for inexpensive inertial sensors as it can be implemented using simple geometric constraints and manual excitation. A novel mathematical framework eliminates the need for precise alignment with external references and creates a set of observation equations that are linear in the parameters to be calibrated. A method of optimizing the set of observations used and results of applying the calibration technique in simulation and with an actual inertial measurement unit (IMU) are presented. The method is shown to be effective in calibrating complex sensor models that account for scale factor nonlinearity, anisotropic sensitivity, and gyro specific-force sensitivity. These models provide an average improvement in the calibration results relative to simple linear models of 5.7% for the accelerometers and 24% for the gyros tested in the IMU. The flexibility of the calibration technique to customize the sensor models and number of observations required according to the desired level of calibration accuracy is also demonstrated and discussed.

4.2 Introduction

Chapter 3 presented a new method for calibrating inertial sensors using simple geometric constraints. While it could be applied to any type of inertial sensor, it was designed for application to consumer-grade sensors based on micro-electromechanical systems (MEMS). A novel mathematical approach allows the calibration to be performed solely using time-averaged and integrated sensor outputs along with easily measured linear and angular displacements. This removes the need for precision equipment capable of precisely controlling the orientation of the IMU relative to external references and for applying known angular rates, as required in traditional laboratory calibration techniques [1, 2].

The new calibration method also enables calibration of complex sensor models that account for scale factor nonlinearities, anisotropic sensitivity, and gyro specific-force sensitivity [1, 2]. For applications where the accuracy of the calibrated sensor outputs must

be maximized, this can offer significant advantages relative to existing magnitude-based calibration techniques and in-use calibration techniques that account only for simple linear sensor models (see Section 3.3). Another advantage of the new calibration method is that the formulation of the observation equations allows the parameters to be estimated using familiar linear least-squares techniques.

In this chapter, several details of implementing the technique are addressed and results demonstrating its application in simulation and with a MEMS-based IMU are presented. The methods used to simulate the calibration of an accelerometer and gyro triad are discussed in Section 4.3. An algorithm to optimize the selection of observations to use in the calibration is then developed in Section 4.4. The results of simulations used to test and apply this algorithm along with simulated calibration results are presented in Section 4.5. Following this, the hardware and data collection methods used to implement the calibration technique with a MEMS IMU are described in Sections 4.6 and 4.7, respectively. The experimental results and a demonstration of the accuracy improvements offered by the various terms included in the sensor models are given Section 4.8. Section 4.9 discusses several points related to the results and implementation of the technique, and concluding remarks are given in Section 4.10.

4.3 Simulation method

4.3.1 Model setup

The models of the sensor output voltages as a function of applied specific force and angular rate defined in Section 3.4 were used to simulate an accelerometer and gyro triad. The parameters were chosen based on the nominal values for the 400°/s gyros and the accelerometers used in the IMU, as described in Section 4.6. To test the efficacy of the calibration method with varying parameter values, the simulated model parameters were randomly selected from the ranges described below.

- Errors in the voltage bias were chosen to be ±0.05–5% of their nominal values.

- Linear scale factors were chosen to have a maximum deviation of ±5% from their nominal values.
- Misalignment angles between the sensor and case axes were chosen between ±1–5°.
- Elements of the anisotropic and squared dependency matrices were chosen such that the impact of each would be between ±0.5–2.5% of the full-scale linear output.
- Elements of the gyro's linear, anisotropic, and squared specific-force dependency matrices were also given impacts between ±0.5–2.5% of the full-scale linear output.
- Tilt angles of the calibration plate relative to the gravity vector were chosen between ±0.1–10°.

With the exception of the scale factors, which would never potentially be zero, minimum magnitudes were enforced to ensure all of the parameters' effects would be meaningful. By doing this, if large uncertainties for the parameter estimates were reported, there was no ambiguity as to whether this was a result of their insignificance or an inadequate set of observations being used in the calibration equations. Since each element of the anisotropic and squared dependency matrices was chosen individually and required to have a minimum impact of ±0.5% of full-scale, their cumulative effect creates a model with significantly more nonlinearity than is expected for even consumer-grade MEMS inertial sensors. While unrealistic, this tested the calibration method's ability to estimate all of the model terms.

4.3.2 Data generation

The sensor outputs for the observations used in the calibration procedure were simulated as discrete-time signals at the same sampling rates used in the experiments described in Section 4.7. Gaussian noise with variance matching that of the outputs from the real sensors was added to the simulated outputs. The simulated tilt angles of the calibration plate were used to obtain the gravity vector in the plate frame. For static observations, this is the only input since Earth's rotation rate is negligible for the gyros used in this study, as discussed in Section 3.7.3.

For the linear motion observations, a sinusoidal acceleration profile was used with appropriate duration and magnitude to cause the desired displacement in a prescribed length of time. This modeled the accelerometer outputs observed in experimental trials very well. The angular rate during the linear motions was modeled as being zero, as in the ideal case, since the experimental results showed no significant response from the gyros for properly executed linear motions.

The finite rotations require both angular rates and accelerations to be simulated since only the direction of the rotation axis is assumed constant. Experimental data showed significant angular rate variation since the motions were performed manually. Rather than approximating these variations, the simulations used a half-sinusoid as the angular rate profile with the amplitude and period set to preserve the intended rotation angle and duration. The duration is the most critical factor given that the data reduction techniques use time-averaged sensor outputs rather than instantaneous values. In the experimental data, the accelerations observed during the rotations included sinusoids caused by changes in the centripetal acceleration as the angular rate varied. The simulated accelerations for the rotations were modeled as sinusoids of similar frequency and magnitude.

4.3.2.1 Speeds and displacements

The maximum accelerations and angular rates for the simulations were determined using the minimum durations in which the linear and rotation motions could be executed accurately in the experiments. Preliminary trials showed that these were approximately 0.3 s for the linear motions and 0.6 s for the rotations. Given the sinusoidal profiles used to generate the simulated data, these resulted in peak accelerations and angular velocities of approximately 1.1g and 300°/s, respectively. These agreed well with the experimental data. While it was possible to move the IMU faster during the experiments, this made it difficult to maintain the correct orientation of the IMU. Therefore, these limits were adopted as it was deemed preferable to have more accurate data than a wider dynamic range.

For the simulated linear motions, the displacements were determined given the maximum travel allowed by the geometry of the hardware. The angular displacements for the rotation observations were fixed at 90° in both the simulations and experiments as trying to execute larger rotations while securing the IMU in a support block proved awkward.

4.3.2.2 Treatment of bias and time windows

Preliminary simulations showed that the regressor's condition number was negatively impacted by large bias values in the sensor models and by using wide time windows that included static data before and after the period of motion in the linear and rotation observations. This is because both factors increase the bias's contribution to the time-averaged and integrated voltage outputs relative to the contribution from the specific force and angular rate inputs.

This problem is illustrated by decomposing the total voltage output from the sensor, $\mathbf{E}_{tot}$, into a constant component due to bias, $\mathbf{E}_b$, and a time- and orientation-dependent component, $\mathbf{E}_{f,\omega}$, due to the specific force and, for the gyros, angular rate inputs according to:

$$\mathbf{E}_{tot}(t) = \mathbf{E}_{f,\omega}(t) + \mathbf{E}_b \ . \tag{4.1}$$

For the sensor biases:

$$\lim_{\mathbf{E}_b \to \infty} \mathbf{E}_{tot}(t) = \lim_{\mathbf{E}_b \to \infty} \left(\mathbf{E}_{f,\omega}(t) + \mathbf{E}_b \right) = \mathbf{E}_b \ , \tag{4.2}$$

meaning that as the bias voltages become very large, the effect of $\mathbf{E}_{f,\omega}$ becomes insignificant in the sensor output. Therefore, in both the simulations and experiments, the voltage used to calibrate and apply the model was:

$$\mathbf{E}_s(t) = \mathbf{E}_{raw}(t) - \mathbf{E}_{b,nom} \tag{4.3}$$

where $\mathbf{E}_{raw}$ is the raw voltage output and $\mathbf{E}_{b,nom}$ is the nominal bias value from the datasheet.

Accurately cropping the time period for linear and rotation observations also improves parameter observability. Including data outside of the motion does not affect the validity of the calibration equations. However, the longer the included stationary periods become, the more any remaining sensor bias contributes to the averaged and integrated output values. Therefore, accurately cropping the data increases the relative influence of the motion on the corresponding rows of the regressor matrix. Experiments showed that it was easy to manually crop the data to within 0.05–0.1 s of the best estimate of the start and end of the motion. Therefore, for the simulations, random amounts of static data in this time range were added at the beginning and end of linear and rotation observations.

The effects of removing the nominal bias and accurately cropping periods of motion on the condition number and minimum singular value of the column-scaled regressor are shown in Table 4.1. The values are the averages from 25 Monte Carlo simulations using the number of observations chosen for the experiments, as identified in Section 4.5.1. For uncropped motions, a random amount of static data between 1–2 s was included before and after the motion. Using the guideline presented in [81, §4.3] that a condition number below 100 generally indicates that there are no poorly identifiable parameters, these results indicate that removing the nominal bias and cropping the motions with reasonable accuracy

Table 4.1: Effects of bias and cropping on observability. c is the condition number and μ_{min} is the minimum singular value for the column-scaled regressor. The method of calculating the values is described in Section 4.3.2.2.

Nominal Bias	Motions Cropped	Accelerometer		Gyro	
		c	μ_{min}	c	μ_{min}
Retained	No	10,300	0.00030	2,030	0.0017
Retained	Yes	390	0.0079	260	0.013
Removed	No	1,500	0.0012	81	0.031
Removed	Yes	35	0.054	11	0.19

are both necessary steps to allow all of the model parameters to be estimated robustly.

4.3.2.3 Simulating orientation errors

Simulations were used to assess the sensitivity of the calibration results to orientation errors that are likely to occur from small geometric errors in the hardware along with errors in manually constraining the IMU's orientation. These errors were simulated by rotating the case frame from its intended orientation about an axis whose direction was chosen randomly from a uniform distribution on the unit sphere. The angle of rotation was also selected randomly from a uniform distribution with a specified maximum magnitude and randomly selected sign. For each observation, the resulting perturbed orientation was used to generate the simulated data. Note that additional sources of experimental error are identified and discussed in Section 4.9.3.

4.4 Observation selection and optimization

This section presents a method of selecting a set of observations that is sufficiently large and diverse to ensure that all of the model parameters can be estimated robustly. This requires special attention for two reasons:

1. The large number of parameters to be calibrated in both the accelerometer and gyro sensor models.
2. The relative importance of the terms in the sensor models may not be known for a previously uncalibrated IMU.

As discussed in Section 4.9.2, the number of observations required in the calibration process can be reduced by eliminating unnecessary parameters from the sensor models. However, in general, it is impossible to know which can safely be removed without first performing a calibration that can accurately estimate all of the parameters. Therefore, it is important to utilize a set of observations that comprehensively spans the available input space in the initial calibration to allow robust estimates of all of the model parameters to be obtained.

To this end, Section 4.4.1 presents an algorithm developed to optimize the linear independence of observations chosen from a set of candidates. Section 4.4.2 then describes its implementation and the process used to determine the number of observations required to ensure good observability for all parameters.

4.4.1 Optimization algorithm

The outputs of the adopted sensor models are determined by the specific force and angular velocity vectors along with their anisotropic products and squared components. These can all be grouped together into a composite input vector:

$$\boldsymbol{\psi} = \begin{bmatrix} {}^{c}\mathbf{f}^{\mathrm{T}} & {}^{c}\mathbf{f}_{ij}^{\mathrm{T}} & {}^{c}\mathbf{f}_{ii}^{\mathrm{T}} & {}^{c}\boldsymbol{\omega}^{\mathrm{T}} & {}^{c}\boldsymbol{\omega}_{ij}^{\mathrm{T}} & {}^{c}\boldsymbol{\omega}_{ii}^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}} \tag{4.4}$$

where the terms follow the definitions used in Section 3.4. Using a set of observations that comprehensively spans the available space of $\boldsymbol{\psi}$ will provide a well-conditioned regressor matrix. Furthermore, the number of observations required can be reduced by maximizing their linear independence.

The following simplifications can be used to define $\boldsymbol{\psi}$ vectors that adequately represent each type of observation:

Static The angular velocity is zero and the specific force can be set equal to gravity.

Linear The angular velocity is zero and the average acceleration during the first half of the motion can be combined with gravity to obtain the specific force.

Rotation The average angular velocity can be used and the specific force vector can be set equal to gravity since there is no intended acceleration during the rotations.

Given the set of $\boldsymbol{\psi}$ vectors representing all of the available candidate observations, they can be stacked as:

$$\boldsymbol{\Psi} = \begin{bmatrix} \boldsymbol{\psi}^{[1]} & \boldsymbol{\psi}^{[2]} & \cdots & \boldsymbol{\psi}^{[m]} \end{bmatrix}^{\mathrm{T}} \tag{4.5}$$

where $\boldsymbol{\psi}^{[i]}$ is the vector for observation i and m is the number of observations in consideration. The linear independence of the observations represented by the rows of $\boldsymbol{\Psi}$ can then be assessed using their dot product after applying two normalizations:

1. The columns of $\boldsymbol{\Psi}$ are normalized to have unit length giving $\boldsymbol{\Psi}'$. This compensates for the different magnitudes and units of the terms in $\boldsymbol{\psi}$ and balances their relative contributions in the dot products.
2. The rows of $\boldsymbol{\Psi}'$ are then normalized to have unit length. The magnitude of the dot products of the rows then depends only their direction and not their length.

Denoting the result of these normalizations as $\boldsymbol{\Psi}''$, the smaller the magnitude of the dot product of two rows of $\boldsymbol{\Psi}''$, the more linearly independent they are. While $\boldsymbol{\Psi}''$ is based on the input vectors, the linear independence of its rows is directly correlated to that of the corresponding rows of the regressor matrix. This leads to Algorithm 4.1 for optimizing the elements and order of a set of observations to maximize the improvement in parameter observability with each successive observation.

An improvement for Algorithm 4.1 is to use a more rigorous method of dealing with ties in the else clause beginning on line 14. Rather than picking a random observation from the list of ties, the second largest magnitudes of their dot products can be compared, followed by the third largest, and so on, until a unique minimum is found. This approach is not shown in the pseudocode to avoid distracting from the main goal and because, in practice, the performance improvement from this refinement was found to be very small.

4.4.2 Implementation

The first step in applying Algorithm 4.1 is to define a list of candidate observations. The geometry of the IMU's case and available support blocks determine the possible case frame orientations. Each orientation can provide a single static observation along with linear and rotation observations at different speeds. These speeds should be selected appropriately given the dynamic range of the sensors and the method of generating the motions. Practical limits when moving the IMU by hand are discussed in Section 4.3.2.1. Given these, four

Algorithm 4.1: Optimize selection and ordering of observations.

Initialize $\mathbf{\Psi}$ using the list of candidate observations, C
Normalize the columns then rows of $\mathbf{\Psi}$ to produce $\mathbf{\Psi}''$
Initialize the list of observations used, U, as the first observation in C or a predetermined list
Initialize a list of remaining observations as: $R = C \setminus U$
while length $(R) > 0$ **and** length $(U) <$ *maxToKeep* **do**
    for $i = 1$ **to** length (R) **do**
        for $j = 1$ **to** length (U) **do**
            $D(j) \leftarrow \left|\mathbf{\Psi}''_{\mathrm{r}R(i)} \cdot \mathbf{\Psi}''_{\mathrm{r}U(j)}\right|$
        end for
        $M(i) \leftarrow \max(D)$
    end for
    if $\min(M)$ is unique **then**
        $i_{min} =$ index $(\min M)$ // This observation in R is the least parallel to, or most linearly independent from, any observation in U
    else // There is a tie to resolve
        $T \leftarrow \{t \mid M(t) = \min(M)\}$
        $i_{min} =$ random element from T
    end if
    $U \leftarrow U \cup R(i_{min})$
    $R \leftarrow R \setminus R(i_{min})$
end while
return U

target accelerations for the linear motions were considered: $0.1g$, $0.25g$, $0.5g$, and $1g$. These span the available range and were found to provide sufficient variation to give good observability.

For the gyros, a generic set of target angular rates were defined as fractions of the dynamic range of each triad. After optimizing this set, the targets were scaled for each triad. Given the available support blocks, simulations showed that the fractions 1/3, 2/3, $3\sqrt{2}/4$, and $3\sqrt{3}/4$ of the gyros' dynamic ranges were sufficient to ensure good parameter observability. The third and fourth values are greater than 1 to allow higher angular rates when the IMU is in a support block and none of the sensor axes are directly aligned with the rotation axis.

Combinations of the above target speeds and case frame orientations that would cause any sensor axis to saturate were excluded from the candidate observations. The sensor models use the outputs from all three of the triads' axes to determine each component of the output vector. Therefore, if any of the axes is saturated, the observation must be discarded since its output does not accurately represent its response to the input.

The hardware described in Section 4.6 provides a total of 120 different case frame orientations, each of which can be used in a static observation. Given the four linear acceleration targets, each of which can be applied in the positive or negative direction, this gives 960 potential linear motion observations. A total of 196 candidate rotation observations were considered after eliminating those that differ only in their initial rotation about the normal to the calibration plate's surface. This gives a total of 1,276 potential observations.

For the accelerometer calibration, each linear observation provides six equations and static observations provide three (see Section 3.5.3). Thus, using all of the potential observations would result in 6,120 equations to calibrate the 33 parameters per accelerometer triad, giving 185 equations per parameter. Each static, linear, and rotation observation contributes three equations to the gyro calibration. This gives 3,828 possible equations to calibrate the 57 parameters per gyro triad, or 67 equations per parameter. Note that while

the gyros could be calibrated without the linear motions, they provide a greater range of specific force inputs and should be used to improve the observability of the parameters describing the specific-force dependency.

This number of observations is clearly much larger than is necessary or practical. Therefore, simulations were used to determine a sufficient number to ensure good observability for all model parameters. The calibration quality metrics used in this process were the regressor condition number and estimated standard deviation of the model output discussed in Section 3.6. Determining an adequate set of observations is simplified by noting that the accelerometer model parameters are essentially a subset of those in the gyro model. Thus, a set of observations that will calibrate the gyros robustly should also provide good results for the accelerometers. Given this, the following procedure was used to choose the set of observations to use in the calibration experiments:

1. All of the static and rotation observations were retained and used as the initial set in Algorithm 4.1, which was used to optimize the order of the linear motions. Monte Carlo simulations were then used to plot the calibration quality metrics defined above versus the number of linear motions used in the gyro calibration. These results were used to determine the number of linear motions needed to ensure robust gyro calibration results.
2. Using the chosen set of linear motions and still retaining all of the static observations, the order of the finite rotation observations was optimized and the simulations were repeated to assess how many should be retained.
3. The order of the static observations was optimized and the same analysis was used to determine the number to retain.
4. Given the number of static, linear, and rotation observations selected to ensure robust gyro calibration results, simulations were used to verify that all of the parameters in the accelerometer model could also be calibrated robustly.

The results of this process are presented in Section 4.5 along with examples of the simulated calibration results.

The number of linear, rotation, and static observations were determined individually and in that order due to the relative difficulty of executing each type using the hardware and techniques described in Sections 4.6 and 4.7, respectively. Linear observations were the most time consuming; therefore, all of the static and rotation observations were retained while evaluating how many linear observations would be needed to minimize the required number. Likewise, static observations were much faster and easier to execute than rotation observations so all of the static observations were retained while evaluating how many rotation observations should be used. If there were no preference for reducing the number of one type of observation over another, the decision of how many to retain could be made solely based on the optimized order returned by Algorithm 4.1 using the complete set of observations.

4.5 Simulation results

4.5.1 Observation selection

To test the effectiveness of Algorithm 4.1, the complete list of potential observations was used to calibrate first a simulated accelerometer triad and then a simulated gyro triad. Figure 4.1 shows the evolution of the regressor matrix's condition number as an increasing number of the observations are used in the simulated gyro calibration. The line for the

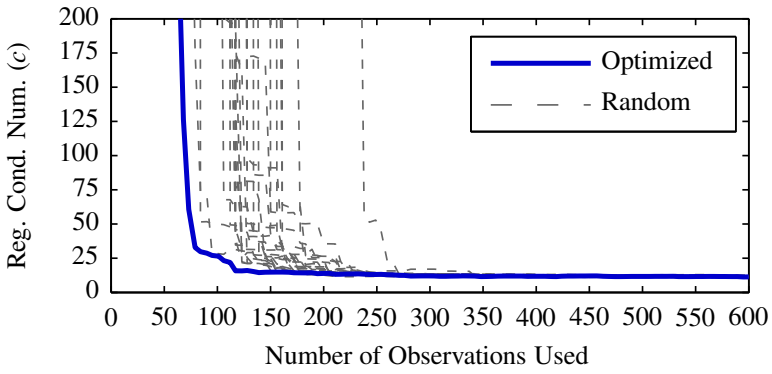


Figure 4.1: The regressor matrix's condition number for calibration of a simulated gyro as the number of observations used increases using the optimized order selected by Algorithm 4.1 compared to 25 randomly selected orders.

optimized order uses the order determined by Algorithm 4.1. For comparison, 25 random orders are also shown. The optimized order provides the fastest improvement up until the condition number has stabilized near its minimum value of 11.6. The value has essentially reached the minimum at around 125 observations using the optimized order whereas the majority of the random orderings require 175–225. This demonstrates that the algorithm orders the observations such that the regressor's condition number and, hence, parameter observability will improve as quickly as possible.

Figure 4.2 shows simulated results for varying the number of linear motions used in calibrating the gyros. The source of the specific force data used in the calibration was a simulated accelerometer calibrated using all of the available observations. The data points in this figure and the remainder of those presented in this section are the averages from 25 Monte Carlo simulations with sensor model parameters selected as described in Section 4.3.1. The two calibration quality metrics plotted are:

1. The regressor matrix's condition number, c.
2. The estimated standard deviation of the calibrated model's output error, ν, divided by the minimum value obtained for that simulation, ν_{min}. To ensure a fair comparison, the residual errors from all of the available observations were used to calculate this metric even when only a subset of the observations were used in the calibration.

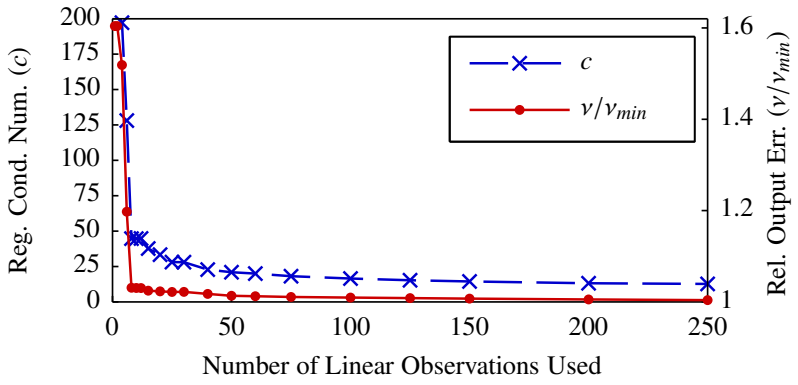


Figure 4.2: Calibration quality metrics vs. linear observations used to calibrate simulated gyros with all of the available static and rotation observations.

Based on the result in Figure 4.2, it was decided to use 50 linear observations. Beyond this point, the rate of improvement of the condition number is very slow and the output error only improves by 1%.

Using this number of linear observations, Figure 4.3 shows the evolution of the same calibration quality metrics as the number of included rotation observations is varied. The condition number reaches its minimum after approximately 40 rotation observations. Unfortunately, more than 100 observations are necessary before the output error suddenly drops to what is essentially its minimum value. Including this many observations would not be practical when performing the calibration by hand. This is especially true given that this number would be required for each gyro triad with a different dynamic range.

The data in Figure 4.3 show each observation near the drop that occurs after 100 observations. There are only two observations that cause the sudden improvement in the error metric, suggesting that those convey important information for improving the results. Indeed, manually moving those to the front of the optimized list causes the drop in the error metric to occur around 40 observations instead of 100. This causes concerns about the optimality of Algorithm 4.1; however, recall that it was designed to maximize the improvement in the regressor's condition number with each successive observation. Looking at that metric, the algorithm is succeeding in its goal. Given these considerations, and the fact that

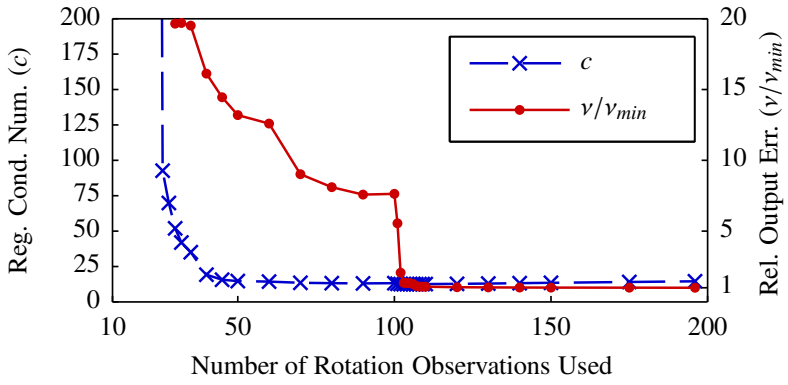


Figure 4.3: Calibration quality metrics vs. rotation observations used to calibrate simulated gyros with all available static observations and 50 linear.

the sensor models used in these simulations are significantly more nonlinear than would be expected for real sensors (see Section 4.3.1), the unusual behavior in the model output error is likely a peculiarity of the simulations and only 40 rotation observations were used in the calibration experiments without modifying the order obtained from Algorithm 4.1.

Given the selected number of linear and rotation observations, Figure 4.4 shows the quality metrics as the number of static observations is varied. Noting the scale of the condition number's axis, it can be seen that the static observations have little impact on this metric given the selected number of linear and rotation observations. Including all of the static observations also only improves the output error by 6% relative to the value if none are used. However, since static observations can be executed faster and more accurately than linear or rotation observations, it was decided to retain 60 static observations. By this point, the majority of the available improvement in both calibration quality metrics has been realized.

To summarize, the number of selected observations was 50 linear, 40 rotation, and 60 static for a total of 150. For the gyros, this provides 450 equations, giving 7.9 equations per parameter to be calibrated. The regressor's condition number using this set of observations is 11.2, which is comparable to the value obtained using all of the available observations.

To verify the assumption that the set of observations chosen for the gyros would also provide robust calibration results for the accelerometers, Figure 4.5 shows the evolution

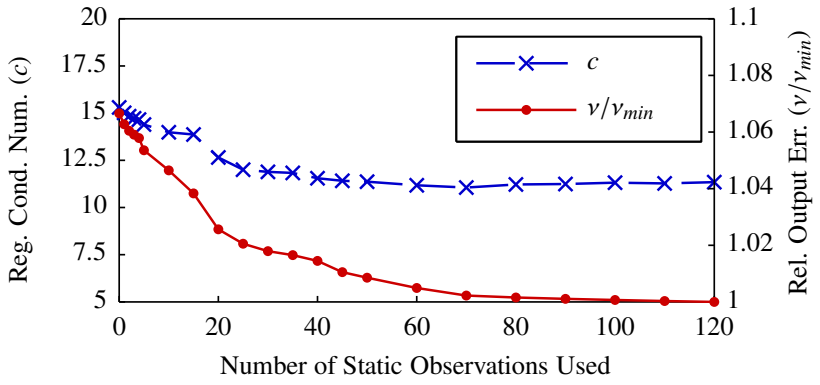


Figure 4.4: Calibration quality metrics vs. static observations used to calibrate simulated gyros with 50 linear and 40 rotation observations.

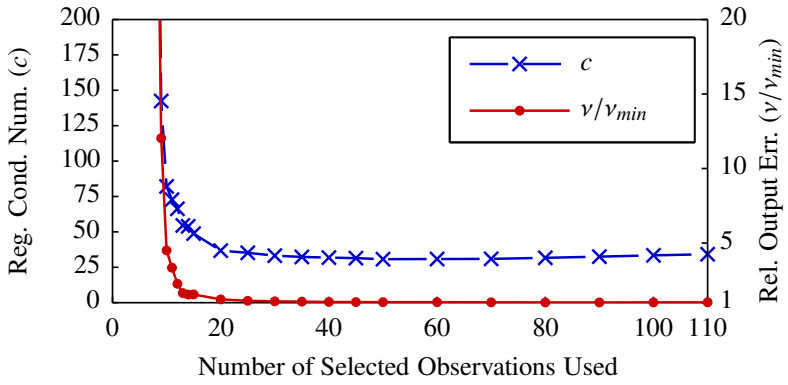


Figure 4.5: Calibration quality metrics vs. selected observations used to calibrate simulated accelerometers relative to the minimums for the complete set.

of the calibration quality metrics for simulated accelerometer calibrations. The reference value for the output error is the result using all of the candidate static and linear observations. The results show that the selected observations provide error and observability metrics equivalent to using the full set of observations.

The 50 linear and 60 static observations will give a total of 480 equations for calibrating the accelerometers. This provides 14.5 equations per unknown parameter and gives a condition number of 34.2. While approximately three times higher than the condition number for the gyro calibration, this is still well below the threshold of 100 that would indicate concern over parameter observability [81, §4.3].

4.5.2 Sample simulated calibration results

Example results for the calibrated parameters of a simulated accelerometer are shown in Table 4.2. The terms are defined as in Section 3.4. Given their similarity to these results, calibrated parameters for a simulated gyro calibration are not shown. The impact columns quantify the significance of the model parameters and their estimated standard deviations. For the bias terms, the impact is the parameter value or its standard deviation over the nominal dynamic range of the sensor. For the remaining terms, it is calculated using the voltage corresponding to the sensor's full-scale output according to the nominal scale factor

Table 4.2: Example results for the calibration of a simulated accelerometer. σ_i is the standard deviation for the estimate of the corresponding parameter, $\hat{\phi}_i$. The method of calculating the impact of each term is described in Section 4.5.2.

Parameter (units)	Calibrated Values	Impact of $\hat{\phi}_i$ (% of Full Scale)	Impact of σ_i (% of Full Scale)
${}^{c}\mathbf{T}_a \left(\frac{\mathrm{mm/s^2}}{\mathrm{V}}\right)$	$\begin{bmatrix} 15050 & 279.2 & 815.5 \\ 605.8 & -14880 & 909.4 \\ 236.1 & 579.1 & 14110 \end{bmatrix}$	$\begin{bmatrix} 101 & 1.88 & 5.49 \\ 4.08 & 100. & 6.12 \\ 1.59 & 3.90 & 95.0 \end{bmatrix}$	$\begin{bmatrix} 0.0590 & 0.00910 & 0.00906 \\ 0.00974 & 0.0596 & 0.00916 \\ 0.00959 & 0.00929 & 0.0559 \end{bmatrix}$
$\mathbf{A}_a \left(\frac{\mathrm{mm/s^2}}{\mathrm{V^2}}\right)$	$\begin{bmatrix} 103.5 & 85.45 & -187.0 \\ 187.9 & 117.8 & -272.9 \\ -211.2 & 172.1 & -101.3 \end{bmatrix}$	$\begin{bmatrix} 0.920 & 0.759 & 1.66 \\ 1.67 & 1.05 & 2.42 \\ 1.88 & 1.53 & 0.900 \end{bmatrix}$	$\begin{bmatrix} 0.0407 & 0.0352 & 0.0409 \\ 0.0406 & 0.0351 & 0.0411 \\ 0.0407 & 0.0353 & 0.0408 \end{bmatrix}$
$\mathbf{S}_a \left(\frac{\mathrm{mm/s^2}}{\mathrm{V^2}}\right)$	$\begin{bmatrix} 93.45 & 170.3 & -115.1 \\ 53.83 & 218.7 & 101.0 \\ -213.4 & 100.6 & 71.42 \end{bmatrix}$	$\begin{bmatrix} 0.830 & 1.51 & 1.02 \\ 0.478 & 1.94 & 0.897 \\ 1.90 & 0.893 & 0.634 \end{bmatrix}$	$\begin{bmatrix} 0.0386 & 0.0312 & 0.0314 \\ 0.0383 & 0.0310 & 0.0314 \\ 0.0382 & 0.0308 & 0.0313 \end{bmatrix}$
$\mathbf{b}_a \left(\frac{\mathrm{mm}}{\mathrm{s^2}}\right)$	$\begin{bmatrix} 695.7 & 186.0 & -362.3 \end{bmatrix}^{\mathrm{T}}$	$\begin{bmatrix} 3.55 & 0.949 & 1.85 \end{bmatrix}^{\mathrm{T}}$	$\begin{bmatrix} 0.00966 & 0.00915 & 0.00913 \end{bmatrix}^{\mathrm{T}}$
Regressor condition number = 31.96 , Estimated model output error standard deviation = $5.948 \frac{\mathrm{mm}}{\mathrm{s^2}}$			

with the nominal bias removed. This voltage, or its square, as appropriate, is multiplied by each parameter and the result is divided by the nominal dynamic range. The result is the contribution of the term to the calibrated model output when the input voltage is equal to the sensor's nominal full-scale output voltage.

Reviewing Table 4.2, the impacts of the standard deviations of the parameter estimates indicate good confidence in all of the parameter estimates, as expected given the low regressor condition number. The influence of the nonlinear terms is also in good agreement with the limits defined in Section 4.3.1.

For the same simulated accelerometer calibration, the calibrated and simulated gravity magnitude, plate frame pitch and roll angles, linear scale factors, and axis misalignment angles are compared in Table 4.3. The accuracy of the calibrated gravity magnitude and plate frame orientation, which are derived from the calibrated components of the gravity

Table 4.3: Comparison of simulated and calibrated values for an example accelerometer calibration. The magnitude of gravity, $\left\| {}^{p}\mathbf{g} \right\|$, is given in mm/s^2, the calibration plate pitch and roll angles, θ_{pitch} and θ_{roll}, are in degrees, the linear scale factors, k_i, are in mm/s^2/V, and the axis misalignment angles, α_{ij}, are in degrees.

Parameter	Simulated	Calibrated	% Error
$\left\| {}^{p}\mathbf{g} \right\|$	9806.6	9797.9	0.0895
θ_{pitch}	−9.900	−9.899	0.00737
θ_{roll}	−1.400	−1.400	0.0145
k_x	15069	15068	0.00369
k_y	−14904	−14892	0.0798
k_z	14154	14164	0.0707
α_{xy}	−1.05	−0.955	9.01
α_{xz}	−2.33	−2.33	0.0257
α_{yx}	2.40	2.34	2.31
α_{yz}	−1.05	−1.06	1.23
α_{zx}	3.48	3.50	0.679
α_{zy}	3.06	3.11	1.67

vector, demonstrates the effectiveness of calibrating these terms. The scale factors and misalignment angles are also in excellent agreement. These were extracted from ${}^{c}\mathbf{T}_{a}$ using the definitions in Section 3.4. A direct comparison of the nonlinear terms is not possible since they do not allow the sensor models to be inverted. Despite this, the linear model terms are expected to be in close agreement, as demonstrated by the results.

4.5.3 Orientation error sensitivity

The sensitivity of the calibration results to orientation errors was quantified using the root-mean-square (RMS) error of ideal observations equations evaluated with parameters estimated using corrupted observations. First, a set of simulated sensors was calibrated without any orientation error using the number of observations selected in Section 4.5.1. The regressor matrix and output vector from this calibration were stored to provide a set of ideal measurements and outputs. Next, the simulated sensors were recalibrated using observations that included random orientation errors generated as described in Section 4.3.2.3. The parameters estimated from the corrupted data were then used with the regressor matrix and output vector from the ideal simulation to calculate the errors for ideal observations.

The RMS value of these errors divided by the RMS value of the errors for the parameters obtained from the ideal calibration is plotted against the maximum simulated orientation error in Figure 4.6. The data shown are the average of 25 Monte Carlo simulations. The graph shows that orientation errors of 0.4–0.5° will cause the RMS error to increase by approximately 50% for both accelerometers and gyros.

These results illustrate that it is important to accurately control the orientation of the IMU during the calibration procedure. However, the sensitivity to these errors is small enough that the hardware and experimental techniques described in the following sections are expected to provide good results relative to the ideal case. This is further discussed in Section 4.9.3.

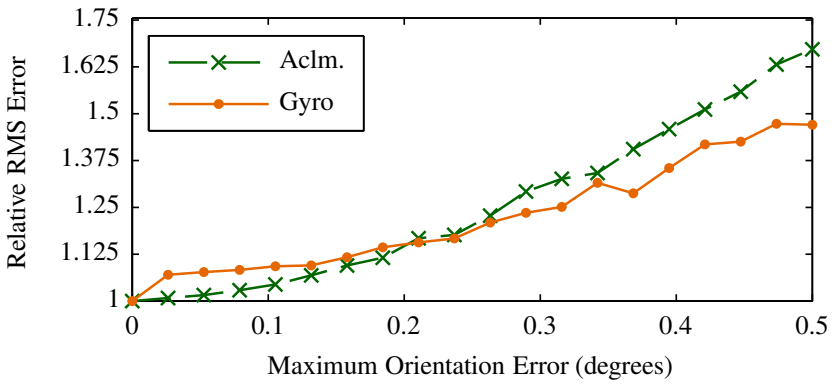


Figure 4.6: RMS error in the calibration equations relative to the ideal case vs. maximum simulated orientation error during each observation.

4.6 Hardware

4.6.1 Inertial measurement unit

The IMU used for the experiments is shown in Figure 4.7. It has two circuit boards mounted back to back with an insulating layer between them. All of the sensors are manufactured by STMicroelectronics. The sensors on the top circuit board are:

- Two LIS344ALH accelerometers offering selectable $\pm 2g$ and $\pm 6g$ dynamic ranges. Only the $\pm 2g$ range was utilized in these experiments.

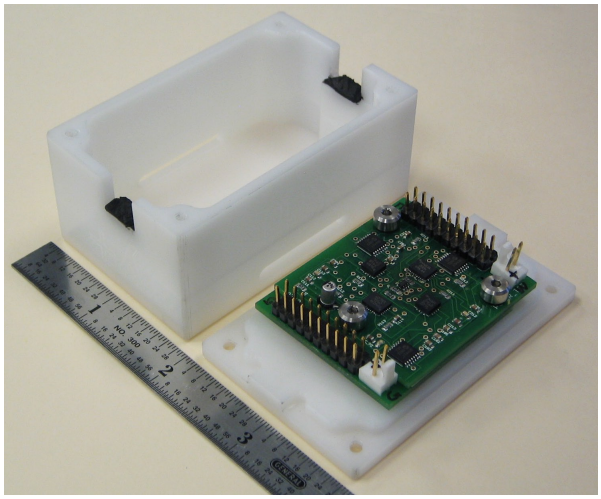


Figure 4.7: The IMU used in the experiments and its open case. A ruler with divisions in inches is shown to provide a sense of scale.

- An LPR403AL pitch/roll and an LPY403AL pitch/yaw gyro. Both provide two output ranges, 30°/s and 120°/s.

The bottom circuit board provides:

- Two additional LIS344ALH accelerometers, which were also fixed in the $\pm 2g$ range for these experiments.
- An LPR410AL pitch/roll and an LPY410AL pitch/yaw gyro. These provide output ranges of 100°/s and 400°/s.

For both circuit boards, the extra in-plane gyro axis provided by the pitch/yaw sensor was ignored for simplicity. The outputs from the remaining axes were grouped by dynamic range and calibrated as four separate gyro triads.

As seen in Figure 4.7, the circuit boards are secured in a case using three screws. Two of the holes in each circuit board were reamed to their final diameter, as was done for the holes in the case. Shoulder screws were then used to mount the boards to provide consistent and accurate alignment.

The case was machined from blocks of acetal using a computer numerical control (CNC) mill. The external dimensions are approximately $47.5 \times 36.5 \times 74$ mm. The final external dimensions were cut with the case assembled to ensure those surfaces would be as planar and orthogonal as possible. The lid includes two exits for the data acquisition cable on opposite faces to allow the IMU to be placed in any orientation in the support blocks and on the plate.

4.6.2 Support blocks and calibration plate

Two support blocks and a calibration plate were also machined for use in the calibration procedure. These are shown with the IMU in Figure 4.8. The V-groove block provides orientations at 45° relative to the plate surface. The compound angle block provides orientation where the three case axes have equal projections on the normal to the plate's surface. It is shown in more detail in Figure 4.9.

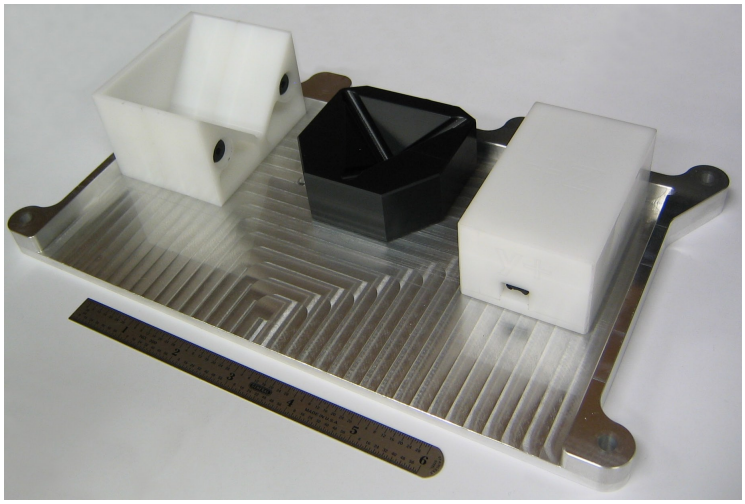


Figure 4.8: The calibration plate (bottom), V-groove block (left), compound angle block (center), and IMU (right).

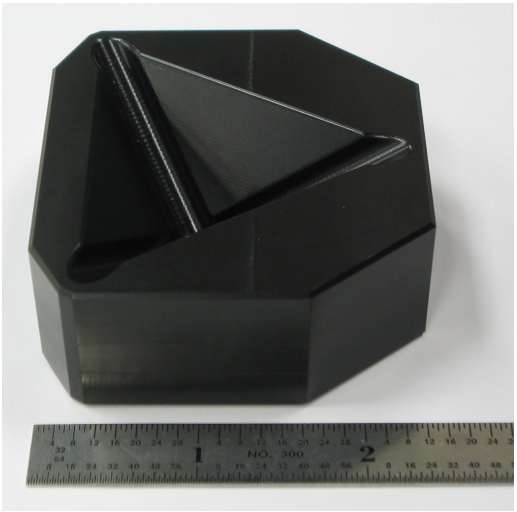


Figure 4.9: Compound angle block.

The calibration plate was machined from an aluminum plate. The lobes on its perimeter have holes that were used to secure it to a tooling plate so that all of the features could be machined in one setup. A screen protector was applied to the main area of the plate and the guide wall to provide a durable, low-friction surface and to prevent tooling marks from causing vibrations as the IMU and support blocks were slid and rotated on the plate.

The V-groove and compound angle blocks were machined from acetal. The V-groove block is made from two pieces held together with shoulder bolts for precise alignment. For each piece, the sides that are perpendicular to the bolts were machined flat and the holes for the shoulder bolts were drilled and reamed. Matching holes in a tooling plate allowed the piece to be secured while the perimeter and V-groove were cut in one setup. Thus, all geometrically important surfaces were accurate within the limits of the mill, specified to be ±0.010 mm positioning accuracy and ±0.005 mm repeatability. After deburring and assembling the two pieces, there is no perceptible error in the coplanarity of corresponding surfaces.

To ensure geometric accuracy of the compound angle block, its bottom surface was milled flat and three holes were drilled and tapped therein allowing it to be placed on a tooling plate and secured from underneath. All remaining surfaces of the block were then cut in one setup. The pocket that supports the IMU was created using a ball-end mill. The grooves seen in Figure 4.9 provide relief for the tool diameter to avoid interference with the corners and edges of the IMU's case.

4.7 Data collection method

To facilitate data collection, the observations were reordered for convenience of execution. This included: (1) ensuring the case would only be opened to switch the cable exit once, (2) grouping static motions together, and (3) minimizing the number of orientation changes required while executing the linear and rotation motions. After particularizing the target angular velocities to the dynamic ranges of the gyros, otherwise identical rotation

observations that differed by less than 20% in their target speed were consolidated to a single observation at the lower speed since attempting to differentiate between very similar rotation rates while moving the IMU by hand is unreliable.

A GUI was implemented to display the intended orientation of the IMU and support block, if used, for each observation. The target angular velocity for finite rotations and acceleration for linear motions was also displayed and the motion could be animated to illustrate the intended speed. Deviations from the target speeds for the motions do not affect the accuracy of the calibration since it uses time-averaged values and displacements rather than instantaneous values. The target accelerations and angular velocities are simply used to help ensure that data throughout the sensors' dynamic ranges are obtained.

For linear motions, the IMU was slid manually across the calibration plate using the guide wall to constrain the motion. The IMU or support block, when used, was started against one of the reference walls of the plate and stopped a short distance from the opposite wall. A digital caliper with a specified accuracy of ±0.02 mm was then used to measure the gap remaining. The displacement was calculated using this value together with the measured dimensions of the IMU case, support blocks, and the distance between the plate's reference walls. It would be more convenient, and the displacement values would be more accurate, if the IMU were moved until it contacted the opposite reference wall. However, experiments quickly showed that no amount of care could reliably avoid impacts that would cause the accelerometer outputs to saturate briefly causing data loss.

The rotation motions were executed by starting with one face of the IMU or support block against a plate wall and rotating until contact was made between another pair of surfaces to ensure the desired angle change. It was most convenient to pivot around an edge of the IMU or support block starting and stopping against the same wall of the plate. Otherwise, it was difficult to avoid impacts that saturated the accelerometers at the end of the motion. Given these considerations, the profile of the compound angle block (see Figure 4.9), which was designed to reduce its weight, made executing the rotations more

difficult. Retaining a rectangular profile would be preferred.

Before beginning the experiments, the IMU was powered and allowed to warm up for approximately an hour to avoid temperature variations during the tests due to self-heating. The sensors' outputs were sampled using a 16-bit ADC with a ±5 V input range. Data for the static observations were sampled at 200 Hz for 5 s. For the linear and rotation observations, a 1000 Hz sampling rate was used. A second GUI window was used to display the data for each observation immediately after its collection. The nominal sensor parameters were applied to the raw data and the resulting outputs were rotated into the plate frame and then plotted. This allowed quick visual identification of problems such as orienting the IMU incorrectly, strong impacts against walls, or unintended orientation changes during an observation. Before saving the data from linear and rotation observations, they were cropped to the period containing the motion, as discussed in Section 4.3.2.2.

4.8 Calibration results

This section presents calibration results for the IMU. First, examples of the calibrated parameters for the IMU's accelerometers and gyros are given in Section 4.8.1. The robustness of the calibration results is then verified in Section 4.8.2. Following this, Section 4.8.3 provides a comparison of the calibrated models' output errors to the sensor noise and output stability. Finally, Section 4.8.4 compares the residual errors for the calibration using different subsets of the parameters from the sensor models to quantify the improvements offered by the various terms.

4.8.1 Calibrated parameters

The magnitude and direction of the estimated gravity vectors obtained from the four accelerometers' calibration results are compared in Table 4.4. The magnitude of gravity used as the truth reference is 9797.7 mm/s^2, as reported by the National Geodetic Survey in June of 1998 at a location a few hundred meters from the building where the tests were conducted. The agreement between the calibrated and reference magnitudes is excellent, with

Table 4.4: Comparison of the results for the calibrated gravity vectors obtained from the four accelerometers in the IMU. The units of the gravity magnitude are mm/s^2, and the angles are given in degrees.

	$\left\|{}^p\mathbf{g}\right\|$	$\left\|{}^p\mathbf{g}\right\|$ % Error	θ_{pitch}	θ_{roll}
Accelerometer 1	9780.	0.182	−0.3762	−0.3839
Accelerometer 2	9793	0.0453	−0.3515	−0.3877
Accelerometer 3	9787	0.114	−0.3628	−0.3826
Accelerometer 4	9780.	0.181	−0.3616	−0.3793
Mean	9785	0.131	−0.3630	−0.3834
Range	13.4	0.137	0.0247	0.00839
Range/Mean × 100%	0.137	NA	6.81	2.19

one of the accelerometers coming within 0.05% of the truth value. The estimated pitch and roll angles of the calibration plate are also in excellent agreement with a maximum discrepancy between estimates from the individual accelerometers of only 0.025° for either angle.

The calibration results for one of the IMU's accelerometers are shown in Table 4.5. The method of calculating the impact of each parameter and its standard deviation is the same as that used for the simulated calibration results (see Section 4.5.2). While the linear scale factors are the most significant terms, as expected, the influence of some of the anisotropic and squared dependency terms is comparable to that of axis misalignment.

Table 4.6 summarizes the calibration results for the 120°/s gyro triad. To evaluate the impact of the terms describing the specific-force dependency, an input of $2g$ was used. This gives a measure of their maximum impact over the working range of the accelerometers. As for the accelerometers, the linear scale factors have the largest impact, but there are several nonlinear terms and specific-force dependencies that have influences similar to that of axis misalignment. Note that the duplication of values in the columns of the matrices giving the impact of the parameter standard deviations is due to the formulation of the regressor matrix. The same data are used in each of the three rows of the gyro's regressor matrix for a given observation. There are also no unknown parameters shared between the three gyro

Table 4.5: Calibration results for one of the accelerometers in the IMU.

Parameter (units)	Calibrated Values	Impact of $\hat{\phi}_i$ (% of Full Scale)	Impact of σ_i (% of Full Scale)
${}^{c}\mathbf{T}_a \left(\frac{\text{mm/s}^2}{\text{V}}\right)$	$\begin{bmatrix} -14820 & 435.5 & 6.360 \\ -60.92 & 14750 & 2.183 \\ -18.05 & 0.2161 & -14770 \end{bmatrix}$	$\begin{bmatrix} 99.7 & 2.93 & 0.0428 \\ 0.410 & 99.3 & 0.0147 \\ 0.121 & 0.00145 & 99.4 \end{bmatrix}$	$\begin{bmatrix} 0.117 & 0.0178 & 0.0169 \\ 0.0177 & 0.117 & 0.0170 \\ 0.0178 & 0.0175 & 0.117 \end{bmatrix}$
$\mathbf{A}_a \left(\frac{\text{mm/s}^2}{\text{V}^2}\right)$	$\begin{bmatrix} 13.07 & 2.869 & -3.338 \\ -3.616 & -5.937 & 24.34 \\ -14.48 & 4.988 & 2.496 \end{bmatrix}$	$\begin{bmatrix} 0.116 & 0.0255 & 0.0297 \\ 0.0321 & 0.0527 & 0.216 \\ 0.129 & 0.0443 & 0.0222 \end{bmatrix}$	$\begin{bmatrix} 0.0736 & 0.0724 & 0.0736 \\ 0.0738 & 0.0722 & 0.0735 \\ 0.0736 & 0.0726 & 0.0735 \end{bmatrix}$
$\mathbf{S}_a \left(\frac{\text{mm/s}^2}{\text{V}^2}\right)$	$\begin{bmatrix} -21.71 & -12.67 & -8.149 \\ -5.649 & 21.30 & 2.456 \\ -12.87 & -16.25 & -34.24 \end{bmatrix}$	$\begin{bmatrix} 0.193 & 0.113 & 0.0724 \\ 0.0502 & 0.189 & 0.0218 \\ 0.114 & 0.144 & 0.304 \end{bmatrix}$	$\begin{bmatrix} 0.147 & 0.138 & 0.144 \\ 0.147 & 0.138 & 0.144 \\ 0.146 & 0.137 & 0.143 \end{bmatrix}$
$\mathbf{b}_a \left(\frac{\text{mm}}{\text{s}^2}\right)$	$\begin{bmatrix} 16.68 & -101.6 & -140.4 \end{bmatrix}^{\text{T}}$	$\begin{bmatrix} 0.0850 & 0.518 & 0.716 \end{bmatrix}^{\text{T}}$	$\begin{bmatrix} 0.0375 & 0.0374 & 0.0372 \end{bmatrix}^{\text{T}}$
Regressor condition number = 30.51 , Estimated model output error standard deviation = $11.85 \frac{\text{mm}}{\text{s}^2}$			

Table 4.6: Calibration results for the 120°/s gyro triad in the IMU.

Parameter (units)	Calibrated Values	Impact of $\hat{\phi}_i$ (% of Full Scale)	Impact of σ_i (% of Full Scale)
${}^{c}\mathbf{T}_g \left(\frac{^\circ/\mathrm{s}}{\mathrm{V}}\right)$	$\begin{bmatrix} 119.6 & -0.2801 & 0.9999 \\ 1.177 & 117.5 & 1.465 \\ 0.2515 & -0.2427 & 120.6 \end{bmatrix}$	$\begin{bmatrix} 99.7 & 0.233 & 0.833 \\ 0.980 & 98.0 & 1.22 \\ 0.210 & 0.202 & 100. \end{bmatrix}$	$\begin{bmatrix} 0.0312 & 0.0298 & 0.0295 \\ 0.0312 & 0.0298 & 0.0295 \\ 0.0312 & 0.0298 & 0.0295 \end{bmatrix}$
$\mathbf{A}_g \left(\frac{^\circ/\mathrm{s}}{\mathrm{V}^2}\right)$	$\begin{bmatrix} -0.01993 & 0.2067 & 0.05673 \\ 0.006555 & 0.06147 & 0.03427 \\ 0.1214 & 0.03004 & 0.1394 \end{bmatrix}$	$\begin{bmatrix} 0.0166 & 0.172 & 0.0473 \\ 0.00546 & 0.0512 & 0.0286 \\ 0.101 & 0.0250 & 0.116 \end{bmatrix}$	$\begin{bmatrix} 0.108 & 0.0774 & 0.0722 \\ 0.108 & 0.0774 & 0.0722 \\ 0.108 & 0.0774 & 0.0722 \end{bmatrix}$
$\mathbf{S}_g \left(\frac{^\circ/\mathrm{s}}{\mathrm{V}^2}\right)$	$\begin{bmatrix} 0.1344 & -0.01006 & 0.03206 \\ 0.1559 & -0.2740 & 0.09537 \\ 0.1356 & 0.07219 & -0.07714 \end{bmatrix}$	$\begin{bmatrix} 0.112 & 0.00838 & 0.0267 \\ 0.130 & 0.228 & 0.0795 \\ 0.113 & 0.0602 & 0.0643 \end{bmatrix}$	$\begin{bmatrix} 0.0797 & 0.0822 & 0.0860 \\ 0.0797 & 0.0822 & 0.0860 \\ 0.0797 & 0.0822 & 0.0860 \end{bmatrix}$
$\mathbf{F}_g \left(\frac{^\circ/\mathrm{s}}{\mathrm{g}}\right)$	$\begin{bmatrix} -0.002742 & 0.01210 & 0.02349 \\ 0.01573 & 0.01830 & -0.006075 \\ -0.01449 & -0.007844 & 0.01404 \end{bmatrix}$	$\begin{bmatrix} 0.00457 & 0.0202 & 0.0391 \\ 0.0262 & 0.0305 & 0.0101 \\ 0.0242 & 0.0131 & 0.0234 \end{bmatrix}$	$\begin{bmatrix} 0.0167 & 0.0169 & 0.0167 \\ 0.0167 & 0.0169 & 0.0167 \\ 0.0167 & 0.0169 & 0.0167 \end{bmatrix}$
$\mathbf{A}_{g,f} \left(\frac{^\circ/\mathrm{s}}{\mathrm{g}^2}\right)$	$\begin{bmatrix} 0.03174 & -0.02468 & -0.009888 \\ -0.01472 & 0.001241 & -0.002126 \\ 0.02191 & -0.0009596 & -0.02588 \end{bmatrix}$	$\begin{bmatrix} 0.106 & 0.0823 & 0.0330 \\ 0.0491 & 0.00414 & 0.00709 \\ 0.0730 & 0.00320 & 0.0863 \end{bmatrix}$	$\begin{bmatrix} 0.0859 & 0.0791 & 0.0828 \\ 0.0859 & 0.0791 & 0.0828 \\ 0.0859 & 0.0791 & 0.0828 \end{bmatrix}$
$\mathbf{S}_{g,f} \left(\frac{^\circ/\mathrm{s}}{\mathrm{g}^2}\right)$	$\begin{bmatrix} 0.01033 & 0.01140 & -0.001506 \\ -0.02621 & -0.002890 & -0.01905 \\ -0.04856 & -0.04577 & -0.03118 \end{bmatrix}$	$\begin{bmatrix} 0.0344 & 0.0380 & 0.00502 \\ 0.0874 & 0.00963 & 0.0635 \\ 0.162 & 0.153 & 0.104 \end{bmatrix}$	$\begin{bmatrix} 0.189 & 0.180 & 0.187 \\ 0.189 & 0.180 & 0.187 \\ 0.189 & 0.180 & 0.187 \end{bmatrix}$
$\mathbf{b}_g$ (°/s)	$\begin{bmatrix} 1.573 & 0.6505 & 0.9910 \end{bmatrix}^{\mathrm{T}}$	$\begin{bmatrix} 1.31 & 0.542 & 0.826 \end{bmatrix}^{\mathrm{T}}$	$\begin{bmatrix} 0.0470 & 0.0470 & 0.0470 \end{bmatrix}^{\mathrm{T}}$
Regressor condition number = 24.41 , Estimated model output error standard deviation = 0.08088°/s			

axes, as with the gravity terms in the accelerometer model. Therefore, the same input data are used to determine the parameters for all three axes and only the values in the output vector differ. This causes corresponding parameters to have equal standard deviations for each axis.

4.8.2 Robustness of the calibration results

As seen in Tables 4.5 and 4.6, the condition numbers for the accelerometer's and gyro's regressor matrices were 30.51 and 24.41, respectively. Both of these are well below 100, indicating that there should not be any significant problems with parameter observability, as discussed in Section 3.6.1. This is reflected in the impact of the standard deviations of the parameters compared to the full-scale outputs of the sensor models. For the accelerometers, the maximum impact of any parameter's standard deviation is 0.147% of the full-scale output. For the gyros, this value is 0.189%. These results show that the maximum uncertainty in the significance of any of the calibrated parameters is well below 1% of the sensor's dynamic range.

Figure 4.10 shows the regressor condition number and estimated output error for the same accelerometer highlighted in Table 4.5 versus the number of applicable observations used in the calibration. It can be seen that there would be no significant change in either

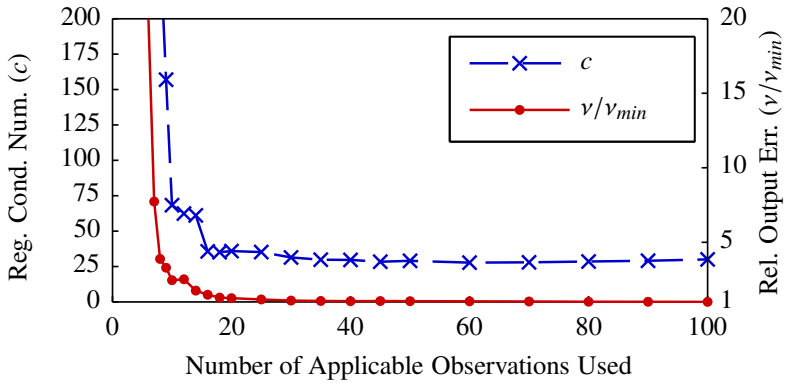


Figure 4.10: Calibration quality metrics, as defined in Section 4.5.1, vs. observations used to calibrate one of the IMU's accelerometers.

calibration quality metric using as few as one third of the available observations. The same metrics are plotted for the 30°/s gyro triad in Figure 4.11. Its results are presented since it saturates most easily. Because of this, the fewest number of rotation observations could be utilized in its calibration and it is the most likely gyro triad to have problems with parameter identifiability. Despite this, the graphs show that around half the number of observations could be used with no significant reduction in the calibration quality.

As mentioned in Section 3.6.1, singular value decomposition can also be used to verify that all of the parameters were estimated accurately. If the singular value decomposition of the column-scaled regressor matrix, $\widetilde{\mathbf{M}}'$, is expressed as:

$$\widetilde{\mathbf{M}}' = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^{\mathrm{T}}, \tag{4.6}$$

where $\boldsymbol{\Sigma}$ is a diagonal matrix of the ordered singular values, then the columns of the orthogonal matrix $\mathbf{V}$ corresponding to the smallest singular values can be used to identify the parameters that are the most sensitive to numerical errors [81,94]. This is illustrated by expressing the vector of parameter estimates, $\hat{\boldsymbol{\phi}}$, in terms of the singular value decomposition as [79, §6.3]:

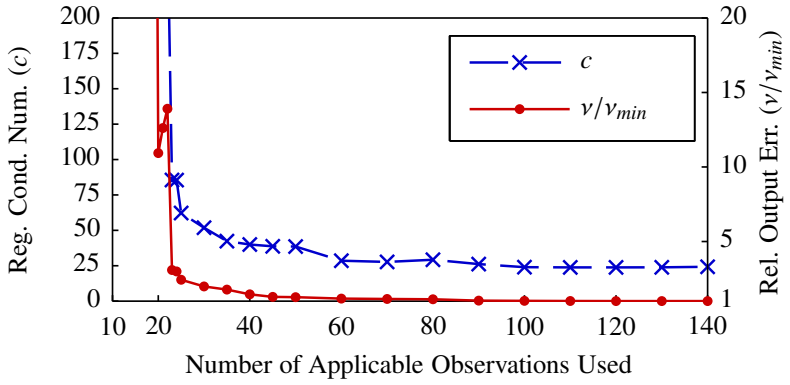


Figure 4.11: Calibration quality metrics, as defined in Section 4.5.1, vs. observations used to calibrate the IMU's 30°/s gyro triad.

$$\hat{\boldsymbol{\phi}} = \mathbf{H} \sum_{j=1}^{n} \frac{\mathbf{U}_{cj}^{\mathrm{T}} \tilde{\mathbf{u}}}{\mu_j} \mathbf{V}_{cj} \tag{4.7}$$

where $\mathbf{U}_{cj}$ and $\mathbf{V}_{cj}$ denote column j of the $\mathbf{U}$ and $\mathbf{V}$ matrices, respectively, μ_j is the corresponding singular value, $\tilde{\mathbf{u}}$ is the output vector for the regression, as defined in (3.42), and $\mathbf{H}$ is the diagonal matrix of column norms used to obtain $\widetilde{\mathbf{M}}'$, as defined in (3.44). Hence, very small singular values will amplify numerical errors in the regressor matrix and output vector. This will, consequently, perturb the estimated values of the parameters corresponding to the rows of $\mathbf{V}_{cj}$ with the largest values.

The singular values for the column-scaled regressor matrix for the accelerometer whose calibration results are shown in Table 4.5 are compared in Figure 4.12. Each is scaled by the maximum singular value and then this quantity is inverted. The result is a measure of the relative amplification of numerical errors for each singular value. The ratio for the smallest singular value corresponds to the regressor's condition number and indicates that some of the parameter estimates are 30.51 times more sensitive to numerical errors than the most robust estimates.

The values from each row of the last column of $\mathbf{V}$, which corresponds to the smallest singular value, are plotted in Figure 4.13. The parameters corresponding to the four

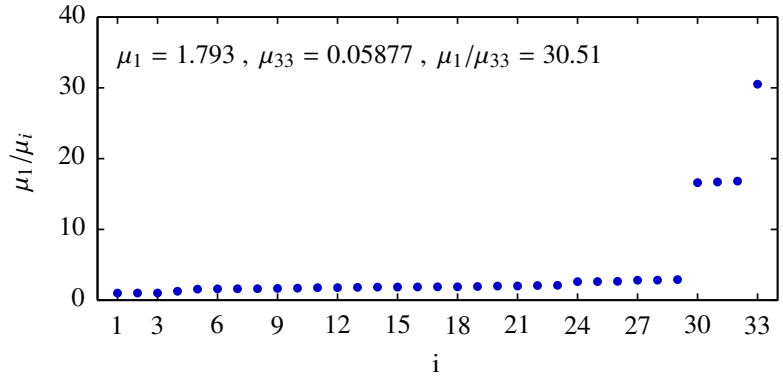


Figure 4.12: Comparison of the singular values for the accelerometer's column-scaled regressor matrix.

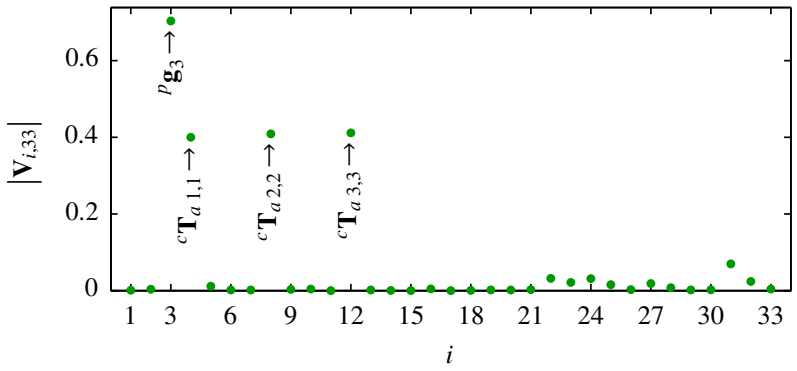


Figure 4.13: Magnitudes of the elements of the column of **V** corresponding to the minimum singular value for the accelerometer's calibration. The parameters corresponding to the most significant elements are indicated.

dominant elements are indicated and are the z component of gravity, ${}^p\mathbf{g}_3$, and the main diagonal elements of the combined scale factor and misalignment matrix, ${}^c\mathbf{T}_a$. Given the small misalignment angles present in the IMU, these elements are very nearly equal to the linear scale factors for each axis. If it were reasonable to believe that one or more of these parameters was unnecessary for the sensor model, they could be removed to improve the regressor's condition number and the observability of the remaining parameters. In this case, this would clearly be inappropriate since the z component of gravity and the linear scale factors are the most significant terms in the adopted model.

Since none of the parameters identified as having the lowest observability can be eliminated from the model, it is prudent to verify that their estimates are accurate and stable by the end of the calibration. In particular, the inclusion of one of the components of the gravity vector in this set could indicate that calibrating it simultaneously with the parameters from the actual sensor model detracts from the overall accuracy of the calibration. One strong indication that this need not be a concern is the accuracy of the magnitudes of the calibrated gravity vectors obtained independently from each of the four accelerometers, as shown in Table 4.4. The excellent agreement in the four estimates of the pitch and roll angles for the calibration plate also indicates that the gravity vector was calibrated accurately.

As an additional verification that the parameters implicated in Figure 4.13 have been

identified robustly, Figure 4.14 illustrates the stability of the four parameter estimates as the number of observations used to calibrate one of the accelerometers is varied. The plotted value is the maximum difference in the impact of any of the four parameters expressed as a percent of the full-scale sensor output compared to the values obtained using all of the available observations. The method of calculating the impact of the components of ${}^{c}\mathbf{T}_{a}$ is defined in Section 4.5.2. For ${}^{p}\mathbf{g}_{3}$, the estimated value is simply divided by the nominal dynamic range of the accelerometer. The plot shows that beyond 40 observations, the maximum difference in the impact relative to the final estimates is less than 0.25%. After 75 observations have been used, there is less than 0.02% variation in the impact of any of the four parameter estimates. This indicates that the chosen set of observations contains sufficiently accurate and varied information that even the parameters that are most sensitive to numerical errors converge to stable, robust estimates.

The same analysis was used to assess the robustness of the parameters in the calibrated gyro model. Figure 4.15 compares the singular values for the 30°/s gyro's column-scaled regressor matrix. The values come in triplets for the same reason that the estimated standard deviations of like parameters for the x, y, and z axes were identical. Namely, the regressor is composed of three copies of the same data and only the values in the output vector differ

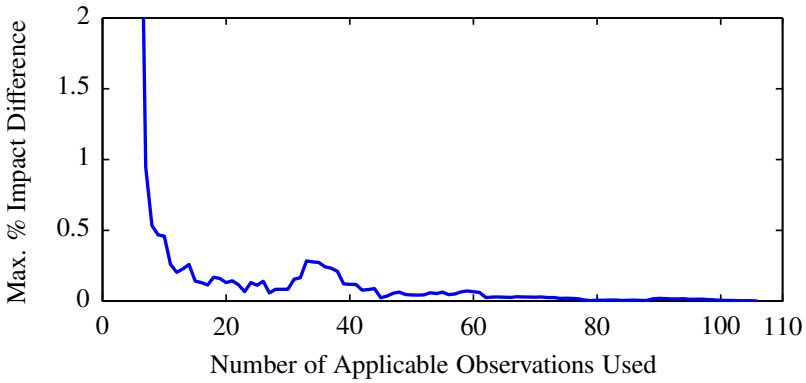


Figure 4.14: Maximum difference in the percent impact of the estimated values for the least-identifiable parameters in the accelerometer calibration compared to the results obtained using all of the applicable observations vs. the number of observations used.

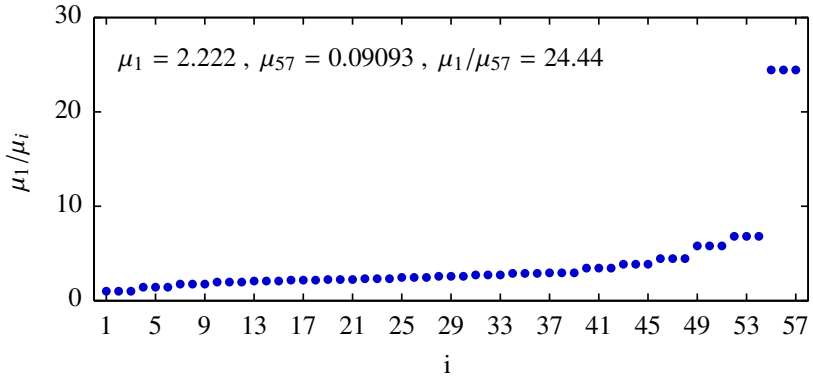


Figure 4.15: Comparison of the singular values for the 30°/s gyro's column-scaled regressor matrix.

for the different axes. As shown, the maximum difference in the numerical sensitivity of the parameter estimates, as indicated by the singular values, is only 24.44.

The elements of the column of $\mathbf{V}$ corresponding to one of the gyro's three smallest singular values are shown in Figure 4.16. The first component of the bias vector, $\mathbf{b}_g$, and the first row of the squared specific-force dependency matrix, $\mathbf{S}_{g,f}$, are the parameters identified as being the least observable. For the other two singular values sharing the minimum magnitude, the second and third components of $\mathbf{b}_g$ and the corresponding rows of $\mathbf{S}_{g,f}$ are the dominant elements of the matching columns of $\mathbf{V}$. To verify that all nine of these parameters were estimated robustly, Figure 4.17 shows the stability of their estimated

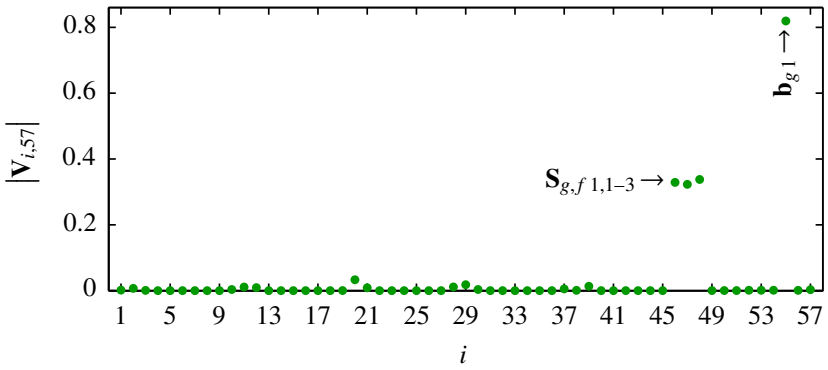


Figure 4.16: Magnitudes of the elements of the column of $\mathbf{V}$ corresponding to one of the three minimum singular values for the 30°/s gyro's calibration. The parameters corresponding to the most significant elements are indicated.

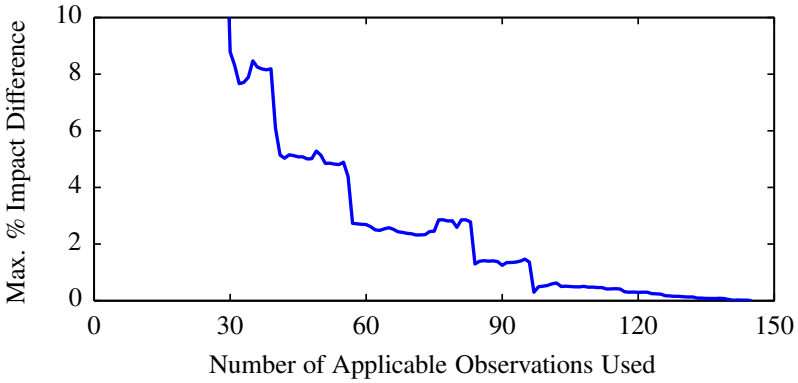


Figure 4.17: Maximum difference in the percent impact of the estimated values for the least-identifiable parameters in the 30°/s gyro calibration compared to the results obtained using all of the applicable observations vs. the number of observations used.

impacts. The maximum difference in the impact relative to the final estimates stabilizes to below 0.6% after 100 observations have been used, and is below 0.08% beyond 135. This indicates that, as for the accelerometers, a very high level of confidence is obtained for even the least-observable parameters in the gyro calibration.

4.8.3 Calibrated model output error

For the accelerometer whose calibration results are shown in Table 4.5, the estimated standard deviation of the calibrated model output error was 11.9 mm/s^2. For comparison, the average standard deviation of the noise present in the output of the triad's three axes was 15.1 mm/s^2, taking the average over all of the static observations used in the calibration. To quantify the stability of the sensor's output, a low-pass filter with a cut-off frequency of 1 Hz was applied to a 10 min sample of static data taken with the IMU at a stable temperature. The range of the filtered data was found to be 22.9 mm/s^2 and its standard deviation was 4.45 mm/s^2.

Similarly for the 120°/s gyro, Table 4.6 shows that the estimated standard deviation of the model output error was 0.081°/s. The average standard deviation of the noise in the three axes' outputs averaged over all of the static observations was 0.168°/s. Using the same filtering technique and 10 min static dataset as for the accelerometers, the range of

the filtered data was 0.181°/s and the standard deviation was 0.025°/s.

These results show that the adopted sensor models reduce the expected errors in the calibrated specific force and angular rate to levels comparable to the sensors' noise and output stability. This indicates that the included parameters compensate for all significant dependencies of the sensor outputs on the applied specific force and angular rate. Further reductions in the model output error would, therefore, require the addition of terms to model the sensors' response to temperature and any other relevant factors beyond the applied motion.

4.8.4 Results for simplified sensor models

Given that one of the key advantages of this calibration method is the ability to calibrate complex sensor models, it is of interest to assess the improvement in accuracy offered by the various terms. To do this, sensor models including only a subset of the terms included in the full model were calibrated using the same set of observations. As discussed in Section 3.7.4, this is done by simply zeroing the terms to be excluded and removing the corresponding columns from the regressor matrix.

Table 4.7 shows a comparison of the model accuracy provided by the nonlinear terms in the accelerometer model. The top row shows that the full model provides an average improvement of 5.7% relative to a linear model accounting for bias, scale factor, and misalignment. The second and third rows show that the squared dependency terms are more significant than the anisotropic sensitivities for the accelerometers used in this IMU.

Table 4.7: Percent increases in the estimated model output error standard deviation, ν, for simplified accelerometer models relative to the results for the full model.

	Percent increase in ν for triad:				
Model Terms Included	1	2	3	4	Avg.
${}^{c}\mathbf{T}_a$, $\mathbf{b}_a$	7.456	5.497	3.617	6.400	5.743
${}^{c}\mathbf{T}_a$, $\mathbf{A}_a$, $\mathbf{b}_a$	5.410	5.017	1.970	5.129	4.381
${}^{c}\mathbf{T}_a$, $\mathbf{S}_a$, $\mathbf{b}_a$	1.707	0.5307	1.808	1.886	1.483

A similar comparison for the gyros is given in Table 4.8 where the influence of both nonlinearities and specific-force dependencies are investigated. Interestingly, the first row shows that the full model provides anywhere from 4.5–70.2% improvement compared to a linear model with bias, scale factor, and misalignment terms. This suggests significant variations in the sensors' linearity and sensitivity to specific force. While the list of potential parameter combinations is not exhaustive, the results in the remaining rows of the table indicate that the most important terms to include for these gyros, in order of significance, are: scale factor nonlinearity, linear specific-force sensitivity, and anisotropic angular rate and specific-force sensitivity.

4.9 Discussion

4.9.1 Accuracy improvements relative to existing techinques

The full set of model parameters considered in this test of the calibration technique provided an average improvement in the estimated model output error of 5.7% for the ac-

Table 4.8: Percent increases in the estimated model output error standard deviation, ν, for simplified gyro models relative to the results for the full model.

	Percent increase in ν for triad:				
Model Terms Included	30°/s	120°/s	100°/s	400°/s	Avg.
${}^{c}\mathbf{T}_g$, $\mathbf{b}_g$	15.80	4.514	70.15	7.047	24.38
${}^{c}\mathbf{T}_g$, $\mathbf{A}_g$, $\mathbf{b}_g$	13.39	3.963	65.11	5.636	22.02
${}^{c}\mathbf{T}_g$, $\mathbf{S}_g$, $\mathbf{b}_g$	3.031	2.990	5.786	5.360	4.291
${}^{c}\mathbf{T}_g$, $\mathbf{A}_g$, $\mathbf{S}_g$, $\mathbf{b}_g$	2.894	2.097	4.961	3.831	3.446
${}^{c}\mathbf{T}_g$, $\mathbf{F}_g$, $\mathbf{b}_g$	14.25	2.914	66.26	4.392	21.95
${}^{c}\mathbf{T}_g$, $\mathbf{S}_g$, $\mathbf{F}_g$, $\mathbf{b}_g$	1.382	1.519	1.872	2.546	1.830
${}^{c}\mathbf{T}_g$, $\mathbf{S}_g$, $\mathbf{S}_{g,f}$, $\mathbf{b}_g$	2.666	2.824	5.123	4.698	3.828
${}^{c}\mathbf{T}_g$, $\mathbf{S}_g$, $\mathbf{A}_{g,f}$, $\mathbf{b}_g$	2.239	2.479	5.162	4.609	3.622
${}^{c}\mathbf{T}_g$, $\mathbf{S}_g$, $\mathbf{F}_g$, $\mathbf{A}_{g,f}$, $\mathbf{b}_g$	0.6219	1.021	1.267	1.856	1.191
${}^{c}\mathbf{T}_g$, $\mathbf{S}_g$, $\mathbf{F}_g$, $\mathbf{S}_{g,f}$, $\mathbf{b}_g$	1.026	1.355	1.407	2.084	1.468

celerometers and 24.4% for the gyros relative to the linear models employed in the majority of existing calibration techniques targeting MEMS inertial sensors (see Section 3.3). This will improve the accuracy of position and orientation estimates obtained by integrating the sensor outputs. For one of the gyros, the output errors were reduced by 70.2%. The resulting reduction in the growth rate of orientation errors will also reduce positioning errors due to inaccurate gravity cancellation, which are currently the dominant source of positioning errors when attempting to navigate using MEMS inertial sensors [37].

As mentioned in Section 4.8.4, the largest improvements in the gyro error were provided by compensating for scale factor nonlinearity and linear specific-force sensitivity. The presence of significant specific-force sensitivity is consistent with results presented in [63]. It is also worth noting that the 30°/s and 100°/s gyro outputs, which exhibited the largest nonlinearity, are actually derived from the same sensor elements as the 120°/s and 400°/s ranges, respectively. This indicates that the amplifier used to provide the more sensitive dynamic ranges in these gyros has significant nonlinearities. The ability of this calibration technique to identify and compensate for these effects is an excellent example of its ability to improve the accuracy attainable with this type of consumer-grade inertial sensor.

4.9.2 Flexibility of the calibration technique

The simulation results in Section 4.5.2 and the experimental results in Section 4.8.2 both demonstrate that the calibration technique is able to provide robust estimates for all 33 parameters per accelerometer triad and 57 per gyro triad for the employed sensor models. However, the experimentally calibrated parameters shown in Section 4.8.1 illustrate that some parameters may be insignificant for a given sensor. As shown in Section 4.8.4, parameters can easily be removed from the models to assess their significance in decreasing the model output error. This allows the complexity of the sensor model to be adapted depending on the desired balance between accuracy and speed of the calibration. Once the appropriate parameters have been eliminated, Algorithm 4.1 and the technique described

in Section 4.4.2 can be used to obtain an optimized set of observations to calibrate the simplified sensor models. With fewer model parameters, fewer observations will be required to obtain robust estimates.

Repeated calibrations of an IMU can also be shortened using the same principles. If certain parameters, such as the components of gravity in the plate frame, are known to be constant between calibrations, their values can be substituted into the calibration equations as known constants. Since these equations are linear in the parameters to be calibrated, the terms associated with the known values can easily be merged into the output vector. The remaining parameters can then be estimated using fewer observations. Aside from the gravity vector, it may be possible to obtain sufficiently accurate estimates of the axis misalignment angles to warrant treating them as constants. Separating the alignment and scale factor matrices and modifying the calibration equations appropriately would then allow only the three scale factors to be treated as unknowns.

4.9.2.1 Hardware variations

Another aspect of the calibration technique that allows significant flexibility is the hardware used in its implementation. As demonstrated here, simple fixtures with a total cost for materials under US$50 combined with a caliper and manual excitation of the IMU allows the technique to be implemented with minimal expense. However, as discussed in Section 4.3.2.1 and Section 4.7, manually executing the motions without any fixed constraints between the IMU and the calibration plate limits the speeds that can be applied and requires a good deal of care to ensure they are executed properly.

For applications where the cost would be justified, more sophisticated hardware and data collection tools would allow the calibration to be completed with greater accuracy, speed, and convenience. For these experiments, after some practice, it took an average of 20–30 s to execute static observations and 30–60 s to execute and crop the data from linear and rotation observations. At this rate, nearly 3 h were required to collect all of the

observations used for the full calibration of the IMU.

Using hardware with linear or rotary joints would reduce or eliminate the need to repeat observations due to errors in constraining the IMU or inadvertent impacts against the walls of the calibration plate. Including a displacement transducer in the linear joint would also eliminate the need to take measurements using the caliper. If desired, a three-axis gimbal mounted on a linear axis would allow the calibration to be completely automated. This would also allow the list of potential IMU orientations to be expanded. The support blocks used in these experiments were designed to provide orientations that would equally excite two or three of the case frame axes. Considering the sensor models to be calibrated and sensitivity analyses presented in [2, 52], these were expected to provide good observability of all of the parameters, as was confirmed in the simulated and experimental results. While not investigated here, using additional orientations could reduce the number of observations required to accurately estimate the parameters.

4.9.3 Sources of experimental errors

The results in Section 4.5.3 showed that IMU orientation errors must be minimized to obtain accurate parameter estimates. With the equipment used in these experiments, errors in manually constraining the IMU and geometric errors in the machined parts are both potential sources of orientation error.

The most significant geometric errors are expected to be the dimensional stability of the materials used and the positioning accuracy of the CNC mill. As described in [73], measurements from a coordinate measuring machine could be used to compensate for geometric errors by correcting the assumed orientations. Such measurements were not performed for these experiments; however, the geometric accuracy of the parts was checked using a dial indicator mounted on the spindle of the mill. When passed over the planar surfaces of the parts, it registered variations on the order of ±0.020 mm, which is twice the machine's specified positioning accuracy. Given the size of the IMU and support

blocks, these variations would cause maximum orientation errors on the order of 0.01°. Therefore, it was assumed that geometric errors in the hardware were negligible relative to other sources of experimental error.

While it is difficult to estimate the orientation errors due to manually constraining the IMU, misalignments as small as 0.25° are visible for the size of the objects involved. Assuming errors around half that size were present, the plots in Figure 4.6 indicate that the degradation of the calibration accuracy relative to the ideal case would be less than 10%.

The displacements for linear and rotation observations are also potential sources of error. Averaging provided by the least-squares solution ensures that while random errors in the displacements will increase the variance of the parameter estimates, their values should not be significantly affected. Systematic errors in the displacements will scale the parameter estimates proportionally since they provide the scale for the observation equations. These errors should be negligible for rotations given the geometric accuracy of the parts. For linear motions, errors in the measured dimensions of the plate, IMU case, or support blocks would cause systematic errors. These would be noticeable over repeated calibrations since the estimated magnitude of gravity would be consistently high or low.

To minimize such errors, multiple measurements of the applicable hardware dimensions were taken and average values were recorded. Then, to estimate the measurement errors likely to occur during the calibration, repeated measurements of the IMU's position on the calibration plate were taken while it was held stationary. The results showed a maximum variation of 0.1 mm, or 0.07% of the average displacement of 144 mm. The accuracy of the calibrated gravity magnitudes shown in Table 4.4 indicates that these displacement measurement errors were small enough to avoid significant errors in the the calibration results.

4.10 Conclusion

The results herein have demonstrated the calibration technique's effectiveness in calibrating complex sensor models using inexpensive equipment and a familiar linear least-

squares solution technique. This makes it accessible to researchers in a variety of fields and allows it to significantly improve the accuracy of calibrated models for inexpensive MEMS inertial sensors relative to existing calibration methods. The technique's flexibility to customize the sensor models based on the characteristics of particular sensors and the desired balance between the accuracy and speed of performing the calibration have also been discussed and demonstrated.

Currently, the main drawback of the technique is the time required for its execution. This is, however, understandable given the large number of parameters calibrated in the sensor models. Future work will include testing the repeatability of the calibrated parameters to identify those that can be considered constant. This will reduce the number of observations and, therefore, the time required to update the remaining parameters in subsequent calibrations. In applications where appropriate equipment is available or the cost of acquiring it would be justified, instrumented linear and rotary axes could also be used to reduce the time required to calibrate the sensors.

Future work will also include developing appropriate methods of compensating for parameter changes due to temperature variations, which were purposefully minimized during these tests. This would reduce errors from environmental temperature variations and self-heating, making data collection more convenient by shortening the warm-up period currently required to let the sensors' temperatures stabilize completely.

4.11 Acknowledgment

The authors would like to thank Amanda Johnson for her assistance and infinite patience while collecting the data used to validate the calibration technique.

CHAPTER 5

ONE-DIMENSIONAL BIAS ESTIMATION AND NOISE REJECTION ALGORITHM

Eric Allen Johnson, Stacy J. Morris Bamberg, and Mark A. Minor

Department of Mechanical Engineering, University of Utah

Salt Lake City, UT 84112

The contents of this chapter are reprinted, with permission,[1] from:

E. A. Johnson, S. J. M. Bamberg, and M. A. Minor, "A state estimator for rejecting noise and tracking bias in inertial sensors," in *Proc. IEEE Int. Conf. Robot. and Autom.*, Pasadena, CA, May 19–23, 2008, pp. 3256–3263.

Formatting and other minor modifications have been applied for consistency with the remainder of this dissertation.

5.1 Abstract

A state estimation algorithm for rejecting noise and tracking bias in both real-time and postprocessing applications of inertial measurement for intermittent motion is presented with experimental results for one-dimensional motion. The algorithm uses characteristics of the sensor noise to automatically recognize motionless periods and update the sensor's bias level without any dependency on application-specific parameters, frequency separation between the signal of interest and the sensor noise, or a high-level system model. Accumulated error in the velocity estimate is eliminated during periods of rest allowing useful velocity estimates to be obtained for arbitrarily long periods given reasonably frequent pauses in the motion. This significantly extends the length of time over which useful position estimates are obtained, especially for postprocessing applications. While derived and demonstrated in the context of inertial measurement, the algorithm can be used to reject noise and/or track bias drift for any signal where zero-level inputs occur intermittently and limits can be set on the noise amplitude, frequency, and bias drift rate.

5.2 Introduction

Inertial measurement technology has been used since the 1920s in aviation with applications in ballistics and space exploration following later in the twentieth century [1]. Throughout this period, a solid knowledge base and sophisticated techniques for inertial measurement have been developed for applications where velocities and distances are measured using nautical miles or even astronomical units [2, 4, 5]. Until recent years, applications better-suited to measurement using meters or millimeters generally did not utilize inertial measurement due to the prohibitive size, weight, and cost of the sensors. The introduction and steady improvement of sensors based on microelectromechanical systems (MEMS) [6–9] and other small, less-expensive inertial sensors [14, 15] has opened the door to their use in such applications in the past decade.

Strapdown inertial measurement units developed from these new sensors offer several

advantages over other position and velocity sensing strategies. Theoretically, acceleration, velocity, position, and orientation information may all be obtained from a single source. This reduces the number of sensors needed and the hardware complexity for the end user. The operation of inertial sensors is self-contained, reducing dependence on outside references, such as global positioning systems (GPS). MEMS inertial sensors are also extremely small and lightweight with tri-axial accelerometers that occupy as little as 15 mm^3 now available [95].

Despite these advantages, applications of inertial sensors are still limited due to drawbacks that include bias drift, sensor accuracy, mounting, and calibration issues. Since velocity, position, and orientation are all derived through integration, the results of which are sensitive to all of these potential sources of error, they lead to unbounded error growth if outside references are unavailable. Still, the advantages of inertial sensors in motion tracking continue to motivate further investigation.

Two applications of interest to the authors that could benefit greatly from the advantages offered by MEMS-based inertial measurement units are human motion tracking and the instrumentation of an ornithopter for flight control. Both of these applications involve motion at relatively low velocities over short distances. Hence, GPS is of very limited utility due to the small range of motion, low speeds, and signal blockage problems associated with indoor environments. Both applications also present limits on the amount of information that can be obtained through a system model. In human motion tracking, the measurement unit could be strapped down in any location on the body and experience a wide variety of motions. For the ornithopter, disturbances caused by even a slight breeze can readily exceed the force generated by the vehicle itself requiring accurate detection of and compensation for unanticipated motions.

While several successful attempts at human motion tracking using inertial measurement have been presented [18, 20, 21], they all rely to some degree on application-specific assumptions and/or external references. Therefore, more general human motion tracking

is still most often and most accurately done using costly vision-based systems. A key drawback of such systems is that the range of motion of the subject is limited to the field of view of the cameras. This often presents significant challenges in gait analysis applications due to space limitations in the motion lab. A successful implementation of inertial measurement techniques could overcome this at a price several orders of magnitude lower than typical vision systems. This potential partially inspired the work described in [19] where the use of a preliminary inertial measurement system was investigated for measuring stride length and other gait parameters. The techniques proved very promising, but the accuracy of the results and ability to implement the technique in real time were limited by the difficulty in tracking the bias level of the inertial sensors. Indeed, the required precision in tracking the bias level of inertial sensors is the principal obstacle in their use for short and long-range applications.

Given these motivations, experiments were conducted to begin the development of real-time bias estimation techniques for short-range inertial measurement applications with reduced dependency on application-specific assumptions. This paper presents a technique that allows real-time estimation of the bias level of an accelerometer and the derived velocity estimate during intermittent, one-dimensional motion. More generally, the algorithm provides real-time noise rejection and bias level estimation in any signal where zero-level inputs occur intermittently and the noise frequency, amplitude, and drift rate are known. The experimental setup and procedures are described in Section 5.3 followed by a description of the noise characteristics observed for the accelerometer output measurement system in Section 5.4. Section 5.5 describes the state estimation technique whose results for the experimental data are shown in Section 5.6. Section 5.7 discusses the capabilities and limitations of the algorithm and a conclusion is given in Section 5.8.

5.3 Experimental setup and procedures

5.3.1 Equipment

The accelerometer used during the experiments was a two-axis, 1.7g Analog Devices ADXL203 MEMS accelerometer. A 5 V supply was used giving a nominal scale factor of 1000 mV/g. The data sheet shows that the value of the scale factor typically varies by less than 2% [96]. Given this accuracy in the scale factor calibration, the nominal value was assumed correct. 100 nF filtering capacitors were used to set the nominal bandwidth of the sensor output at 50 Hz and the nominal noise floor at 1.0 mg̃rms (1.0 mV rms) with a peak-to-peak noise estimate of 6 mg (6 mV) [96].

A THK linear bearing rail and slider were used to constrain motion to one dimension and allow the data processing to focus solely on sensor-level noise rejection and bias estimation without the additional complications of higher-dimensional navigation solutions. It was verified that the precise fit of the slider on the rail constrained their relative motion sufficiently that no measurable changes in the bias level of the sensor could be produced by misalignment between the two. The rail was approximately 800 mm long and was mounted horizontally atop two boards that were nearly coplanar, but which did introduce slight changes in tilt from one end to the other. The maximum variation in tilt along the direction of motion was approximately 0.5°. This variation was large enough to cause measurable changes in the zero level, or bias, of the accelerometer, but small enough that the impact on the scale factor in the direction of motion was negligible. The variation was used to partially simulate the errors in knowledge of the direction of gravity which result when using a gyroscope to track orientation or other conditions that can cause uncertainty in the bias level of the sensor to accumulate faster than by natural drift.

A six-camera Vicon 460 vision system was used to provide an absolute position reference during the experiments for comparison with the results from the accelerometer. Position information from the cameras was provided at a rate of 120 Hz with an accuracy of approximately ±0.5 mm. The Vicon system also provided synchronization of the

accelerometer output and position information through a 16-bit A/D converter which was used with a ±5 V range to sample the sensor output in the direction of motion at 1080 Hz. Due to the distance between the computer system and the field of view of the cameras, the connection between the sensor and the ADC was made using a 15 m ribbon cable that, lamentably, was unshielded.

5.3.2 Experiments performed

First, static measurements were taken in order to accurately characterize the overall noise of the measurement system. Dynamic experiments were then performed by moving the slider back and forth by hand. Different speeds were used along with intermittent motionless periods of various durations and frequencies. In all experiments, the sensor was powered up for a period of at least 10 min to allow a steady-state operating temperature to be reached and avoid bias drift due to transient thermal effects.

5.4 Sensor noise characterization

Representative data from static measurements of the accelerometer's horizontal axis are shown in Figure 5.1–5.3. From the static measurements, the following overall noise characteristics were identified for the measurement system employed:

1. A dominant frequency of 60 Hz. This is believed to be due to the combination of developing problems in the ADC (which only worked intermittently) and the unshielded ribbon cable rather than the sensor itself.
2. A peak-to-peak amplitude of 53.7 mV and standard deviation of 4.79 mV. This is roughly ten times greater than the expected noise level of the sensor alone. The difference is likely attributable to the same factors listed in the previous item.
3. A maximum bias level drift rate of 6.94 mV/s. This value was obtained by averaging the data over successive 2 s intervals and then finding the maximum rate of change from one average to the next.

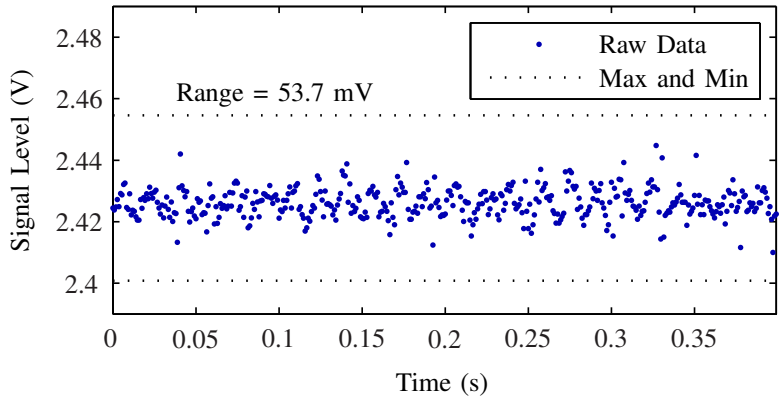


Figure 5.1: Measurement noise during a static test of the accelerometer.

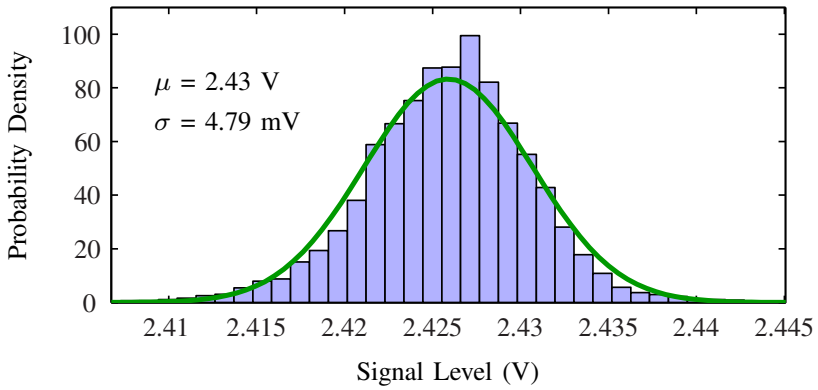


Figure 5.2: Experimentally measured noise distribution compared to the normal distribution derived from its mean, μ, and standard deviation, σ.

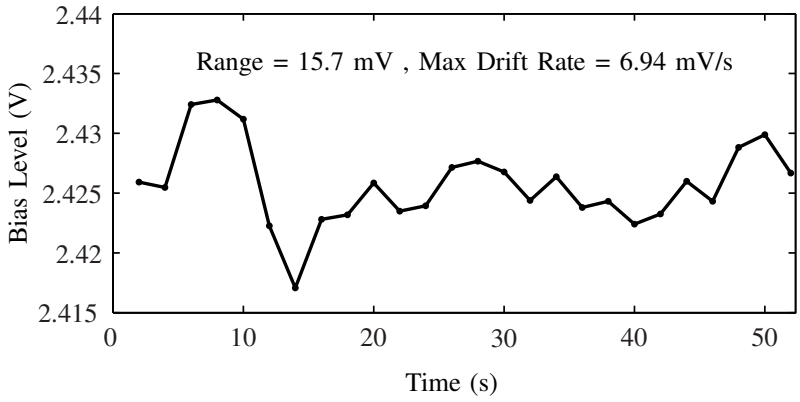


Figure 5.3: Observed bias drift during a static test of the accelerometer. Each point is the average of the data over the prior 2 s.

These values were found to be reasonably consistent throughout the experiments. They are also in very good agreement with the values observed for the vertical axis for which the noise was found to have the same dominant frequency, a peak-to-peak amplitude of 48.7 mV, a standard deviation of 4.55 mV, and a maximum drift rate of 5.2 mV/s.

5.5 State estimation algorithm

5.5.1 Objectives

Three objectives motivated the development of the state estimation algorithm. These were to:

1. Recognize periods where measurement noise dominates any underlying signal to allow the estimate of the sensor's bias level to be updated.
2. Smooth the signal as much as possible without introducing phase lag, overshoot, or otherwise producing errors greater than those already present due to the sensor noise.
3. Minimize processing and data-storage requirements to facilitate using the algorithm in real-time applications.

It should be noted that appropriate data smoothing during periods of nonzero acceleration will have little to no effect on the result of integrating the accelerometer signal to obtain velocity and position estimates. However, it is still useful in that it facilitates the detection of regions where the sensor output is essentially constant, aiding in the first listed objective. Smoothing is also desirable in situations where acceleration is an input to a control algorithm; for example, to produce disturbance force estimates. In such cases, smoothing the accelerometer output will result in smoother control inputs.

5.5.2 Assumptions

First, the assumption was made that the noise characteristics observed in the static tests would remain constant throughout the entire measurement range of the sensor under both static and dynamic conditions. The assumption seems well justified for the static case, and

tests will be performed in the future to see if it is also valid under dynamic circumstances. Holding strictly to this assumption for now, it can be said that:

1. During periods of a constant underlying measurement value, a low-pass filter with a cutoff frequency below the dominant noise frequency must be used to smooth out the noise and recover the true signal value.
2. Any given raw reading will have no more error than one-half the peak-to-peak noise amplitude.
3. It cannot be determined if changes in the signal level at or below the maximum drift rate of the bias level are due to changes in the underlying signal level, drift, or a combination of both.

These observations set realistic limits on the signals that the sensor is capable of accurately resolving. Any signal that falls entirely inside the peak-to-peak noise amplitude and whose mean changes at a rate equal to or below the maximum drift rate is indistinguishable from the inherent noise in the sensor output. For convenience, this range of signals will be referred to as the sensor's noise band. For the sensor to be used successfully, the application must not require measurements in this range. Therefore, a second assumption is made that the sensor is being used prudently and if its output does lie within the noise band, it is because the underlying signal value is constant.

To maximize the applicability of the state estimator algorithm, additional assumptions related to the most general case of inertial measurement were made. Namely, since the measurement unit could be attached to any object, no knowledge of a system model or the frequencies of motion of interest was assumed. Therefore, simple low, high, and band filtering techniques that use frequency information to separate noise from the signal of interest cannot be applied without the risk of attenuating meaningful signals and introducing undesirable amounts of phase lag. Data fusion techniques such as Kalman filtering also cannot readily be applied under these assumptions due to the lack of a system model and/or additional sensors to provide an independent estimate of the states of the system. Other studies have used the Kalman filter at the sensor level with encouraging results [36], but

this requires the adoption of some kind of a system model. A typical choice is a model that assumes zero acceleration at all times. This is not unreasonable in that whatever mass to which the sensor may be attached has inertia that resists acceleration. However, the forces acting on the mass and causing its motion (the quantity of interest) are then viewed as disturbances, creating an interesting irony. Given no knowledge of these forces, the Kalman filter simply acts as a low-pass filter that is unduly difficult to implement.

5.5.3 Algorithm

The algorithm developed to realize the state estimation objectives is shown in the form of a flowchart in Figure 5.4. The structure is similar in spirit to the hybrid switching formulation used in [38]; however, the need for various application-specific thresholds for recognition of motionless periods is eliminated. The major components of the algorithm are described in more detail below.

5.5.3.1 Noise rejection

To reject noise and smooth the sensor output during periods of measurable acceleration, a prediction/correction approach is employed. First, the prior filtered data points are used to estimate the slope of the signal and extrapolate to the present sample to predict what the value would be if the signal continued on its previous course. The raw signal values are not used in this prediction step because they are known to contain noise and would be expected to corrupt the slope estimates because of this. The method of prediction employed was a first-order extrapolation where the slope is estimated from:

$$m[n] = \frac{1}{4}\sum_{i=1}^{4}\frac{1}{i}\Big(y[n-1] - y[n-1-i]\Big), \tag{5.1}$$

where $y[n]$ is the filter output and $m[n]$ is the estimated slope at sample n. This method of estimating the slope is quite simple, yet was found to give a good balance of immunity to

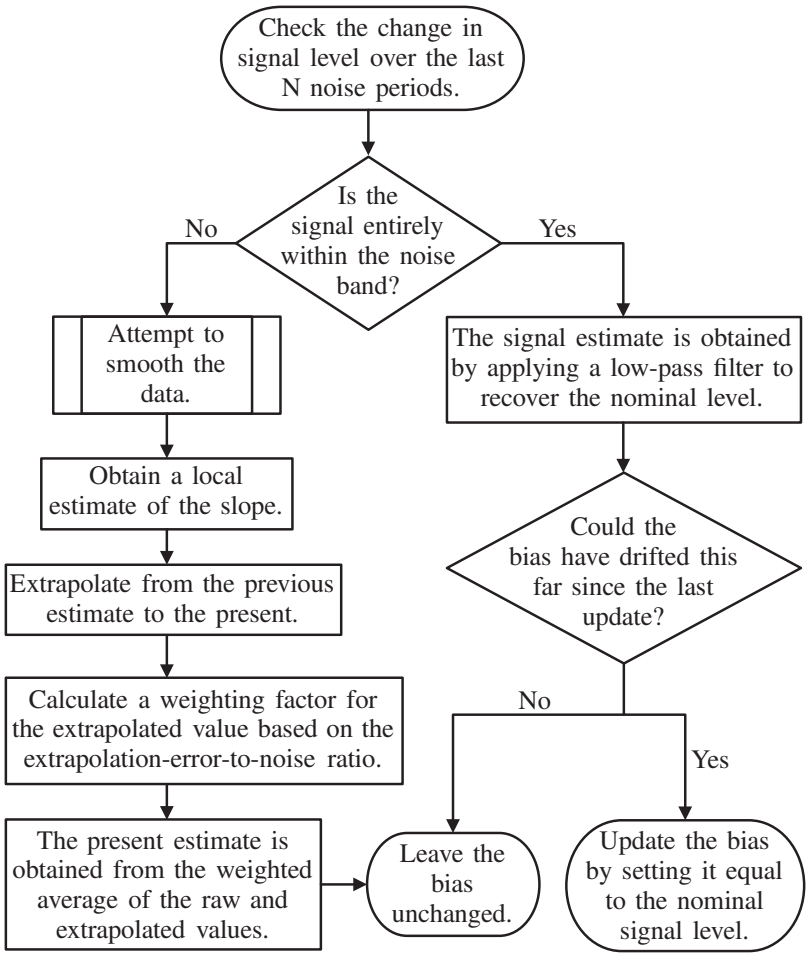


Figure 5.4: Flowchart of the noise rejection and bias estimation algorithm. The processing steps shown are repeated for each incoming data sample.

small fluctuations in $y[n]$ without introducing so much of the data history that it becomes unresponsive. Given this estimate of the slope, the predicted value, $p[n]$, is found simply as:

$$p[n] = y[n-1] + m[n] \tag{5.2}$$

and the ratio of the error between this prediction and the raw sensor reading, $x[n]$, to the peak value (one half the peak-to-peak value) of the noise, η_{max}, is designated $r_{e/\eta}$ and defined as:

$$r_{e/\eta} = \frac{\left|p[n] - x[n]\right|}{\eta_{max}} \,. \tag{5.3}$$

The filtered value is then found from a weighted average of the raw value and the prediction as:

$$y[n] = w(r_{e/\eta})p[n] + \Big(1 - w(r_{e/\eta})\Big)\, x[n] \,, \tag{5.4}$$

where $w(\cdot)$ is a function that weights how much the prediction should be trusted relative to the raw reading and whose range must be restricted to $[0, 1]$ to ensure that $y[n]$ will lie between $x[n]$ and $p[n]$.

To ensure that the smoothing filter never introduces any more error than is already present due to the sensor noise, it is required that:

$$\left|y[n] - x[n]\right| \leq \eta_{max} \,. \tag{5.5}$$

Substituting (5.4) into (5.5) and simplifying gives:

$$w(r_{e/\eta})\left|p[n] - x[n]\right| \leq \eta_{max}\ , \tag{5.6}$$

which can be simplified further via (5.3) to:

$$w(r_{e/\eta}) \leq \frac{1}{r_{e/\eta}}\ . \tag{5.7}$$

Hence, the function $w(r_{e/\eta})$ must be bounded above by both 1 and $1/r_{e/\eta}$.

Given the above restrictions and the nature of the sensor noise, a natural choice for w is that of an unscaled Gaussian distribution of the form:

$$w(r_{e/\eta}) = \exp\left(-\frac{r_{e/\eta}^2}{2\sigma^2}\right). \tag{5.8}$$

This gives a maximum value of 1 when the prediction and raw value are in perfect agreement. This can be viewed as an indication that the weighting function thinks the probability that the prediction is correct is 1. If there is an error between the predicted and raw values, then the function decreases the weight, or the probability that the prediction is correct according to a normal distribution, which was found to be an accurate representation of the sensor noise. The maximum variance that can be used without passing the $1/r_{e/\eta}$ bound is $\sigma^2 \approx 1.6$. This gives the maximum amount of filtering possible for this weighting function while still enforcing (5.5) and was found to give excellent results.

5.5.3.2 Noise band recognition and bias estimation

When the signal is entirely within the noise band, a low pass filter can be used to estimate the underlying signal value. A simple implementation for the filter is a moving average of the form:

$$y[n] = \frac{1}{N\,T_\eta - 1} \sum_{i=0}^{N\,T_\eta - 1} x[n-i]\ , \tag{5.9}$$

where T_η is the number of samples corresponding to the period of the lowest dominant frequency of the sensor noise and N is the number of periods over which the moving average should be taken to ensure that the noise is adequately suppressed. For the accelerometer used in this study, $N = 6$ was found to work well. It must be high enough to smooth the data sufficiently while still being responsive enough to follow potential drift in the bias level.

Since accurate recovery of the average signal value requires averaging samples for N noise periods, the algorithm waits for the output to be entirely within the noise band for that many samples before switching to this mode and using the low-passed values as the filter output. Otherwise, data from a nonstationary signal period will be included in the moving average, shifting the result from the present value.

To recognize when the signal is within the noise band, the raw value at sample n is compared against the filtered value at sample $n - N\,T_\eta$. If the difference is less than the peak noise, then it computes the average of $x[n]$ over the same range and ensures that none of the deviations between $x[n]$ and this average are greater than the peak noise plus the amount the bias could have drifted over that time given the maximum bias level drift rate of the sensor. If all samples are within this range, then the signal is contained within the noise band and the low-pass filtering mode is initiated. While in this mode, the bias level will be updated if the filtered value is within the range to which the bias level could have drifted since it was last updated given its maximum drift rate. For sensors where the bias level is known to stay within certain bounds, these can also be taken into account when deciding if it should be adjusted.

A final consideration is that if the bias level is updated during a stationary period whose duration is extremely short, the resulting value may not be sufficiently accurate. To prevent

this without having to extend the length of the moving average filter, a delay of a specified number of noise periods can be added between the time when the signal is recognized as having entered the noise band and the time when bias adjustment begins. Using a delay of two noise periods was found to work very well in the experiments.

5.6 Experimental results

The data shown in this section are from a 30 s trial in which the slider was moved somewhat randomly with varying velocities. Motionless periods were introduced roughly every 3 s so that every 2 s of motion were followed by 1 s of rest, on average. Similar results were obtained in ten additional trials. The data from this trial are highlighted as they illustrate well both the strengths and weaknesses of the method.

5.6.1 Noise rejection and bias level correction in the accelerometer output

The results of applying the state estimator to the raw accelerometer voltage output are shown for a representative time interval in Figure 5.5. The beginning of the interval is

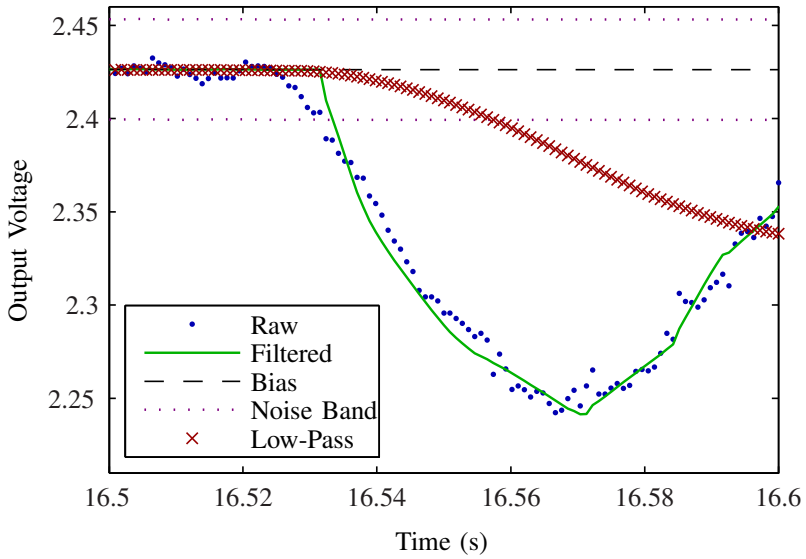


Figure 5.5: Smoothing by the state estimator's method of noise rejection compared to a low-pass filter for the raw accelerometer output.

during a motionless period where the algorithm has identified the signal as being entirely within the noise band and applied the low-pass filter to recover the nominal signal level. This level is within the range to which the bias could have drifted since the last update, so it is updated during this period.

The latter portion of the plot shows that as the signal exits the noise band, the state estimator responds quickly and begins to use the prediction/correction smoothing technique instead of the low-pass filter. The plot shows that the technique is successful in smoothing the signal by rejecting variations that are recognizable as noise. It does this without introducing any significant phase errors regardless of the frequency components of the motion because of the formulation of the weighting coefficient given by (5.8). For comparison, the result that would be obtained if the low-pass filter were used exclusively to process the data is shown. This illustrates the large amount of phase lag that would result due to the low cutoff frequency that is necessary to smooth the signal adequately while it is inside the noise band.

The state estimator's function of recognizing periods where the signal is entirely within the noise band is illustrated in Figure 5.6. When the signal is recognized as having entered the noise band, it is marked as being settled. After remaining settled for the designated number of noise periods, bias level adjustment begins if the signal is within the possible range of drift since the prior update. For the chosen parameters and the noise characteristics of the experimental setup, the delay between settling into the noise band and the beginning of bias adjustment is 0.133 s. The algorithm can reliably recognize and use a stationary period of this short duration to update the bias level of the sensor.

5.6.2 Improvements in accelerometer-derived position and velocity estimates for real-time applications

In a real-time application, each new estimate of the bias level would be used throughout the next period of motion. Proceeding in this fashion and integrating to obtain the velocity estimate, it was found to have a strong drift rate during periods of motion in these

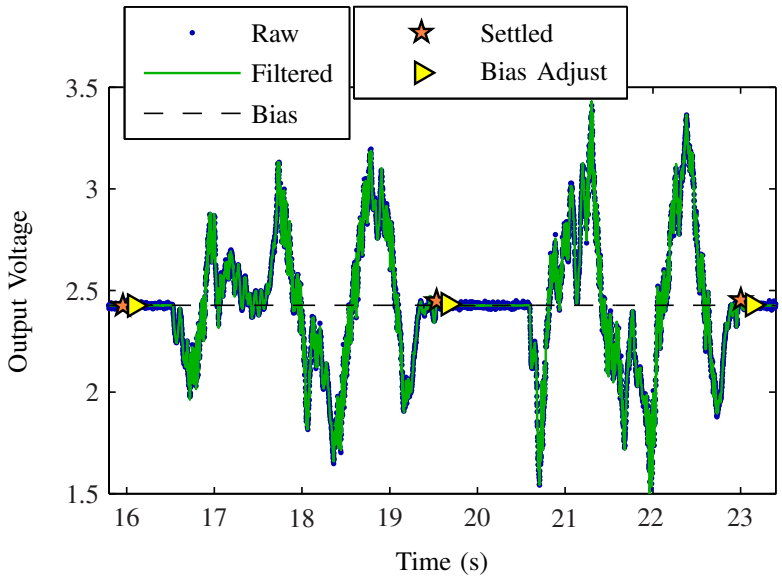


Figure 5.6: Noise band recognition and beginning of bias adjustment periods for the raw accelerometer output.

experiments. The integrated velocity data were examined allowing identification of a noise amplitude, frequency, and maximum drift rate. Using these noise characteristics, the state estimator was applied to the integrated velocity signal to allow its bias level to be estimated. This could also be done in real-time and the result that would be obtained for the same representative time interval is shown in Figure 5.7.

For the velocity estimate, a relatively high maximum drift rate was found (150 mm/s/s). This results in the signal being identified as having settled into the noise band in several unexpected places, but it does not remain there long enough for bias adjustment to begin. This illustrates that when drift rates are high, it becomes ever more difficult to distinguish between meaningful signals and changes due to a potentially drifting bias level.

Shifting the velocity estimate by the updated bias level each time it is revised during a motionless period provides significant improvements over the original estimate obtained with bias corrections for the acceleration only. This is shown in Figure 5.8. The velocity reference obtained by numerically differentiating the data from the vision system is also shown to provide a standard for comparison.

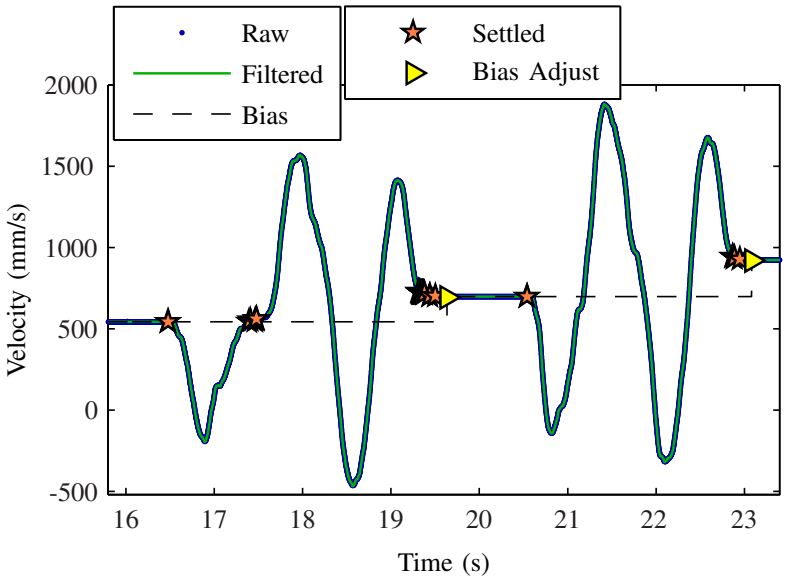


Figure 5.7: Noise band recognition and bias adjustments for the velocity estimate obtained by integration of the processed accelerometer data.

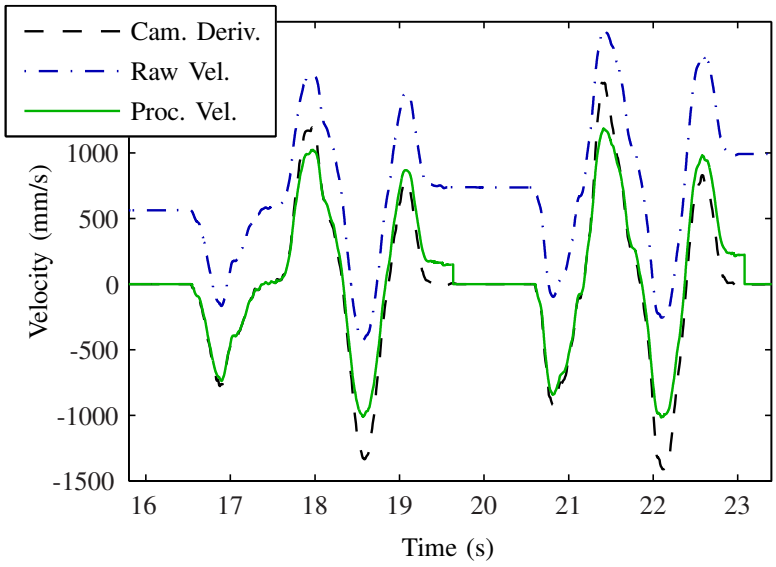


Figure 5.8: Comparison of the velocity estimates derived from the accelerometer with (Proc. Vel.) and without (Raw Vel.) bias correction to the reference provided by the Vicon vision system (Cam. Deriv.).

The position estimate found by integrating the velocity data with and without velocity bias correction is shown in Figure 5.9. At the end of the displayed interval, the error without velocity bias correction is 7065 mm. With velocity bias correction the error is reduced to 1130 mm. While still sizable, this is an 84% reduction. Note that the accelerometer-based quantities do not start at zero because the data shown are taken from the middle of the experiment where the differences in the outcome of the two processing methods are more apparent.

5.6.3 Additional improvements for postprocessing

In postprocessing applications, additional improvements to both the position and velocity estimates may be obtained by linearly interpolating the bias level of both the acceleration and the velocity estimate between the beginning and end of consecutive adjustment periods. This is similar to the approach taken in [20] where the orientation estimate for an inertial pen device is linearly interpolated between start and end values where pauses in the motion are detected using various thresholds developed for the application of character recogni-

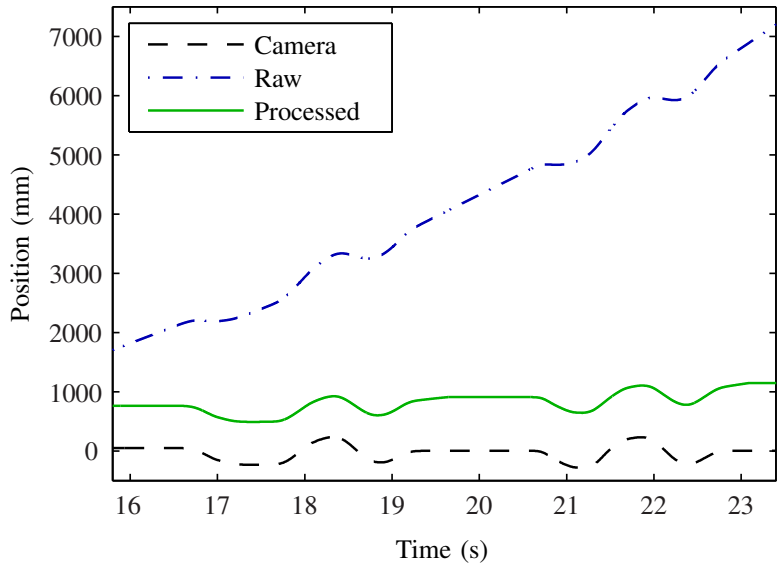


Figure 5.9: Comparison of the position estimates derived from the accelerometer using the velocity estimate with (Processed) and without (Raw) bias correction to the reference provided by the Vicon vision system (Camera).

tion. The result of using this interpolation approach for the velocity estimate is shown for the entire trial period in Figure 5.10. The position estimate obtained by integrating the postprocessed velocity estimate is shown for the entire trial period in Figure 5.11. The maximum error observed during the experiment is only 65 mm compared to a total distance traveled of 8474 mm.

5.7 Discussion

5.7.1 Inertial measurement applications

The state estimation technique presented herein has the limitation that it depends on periods of zero acceleration and zero velocity to update the corresponding bias estimates and correct drift. However, the pauses in the motion need only last a few tenths of a second for bias corrections to occur and the algorithm depends solely on easily measured characteristics of the sensor. This eliminates the need for empirically derived, application-specific parameters and thresholds, which can be difficult to define and tune.

Applications that offer relatively frequent pauses in their motion and to which the

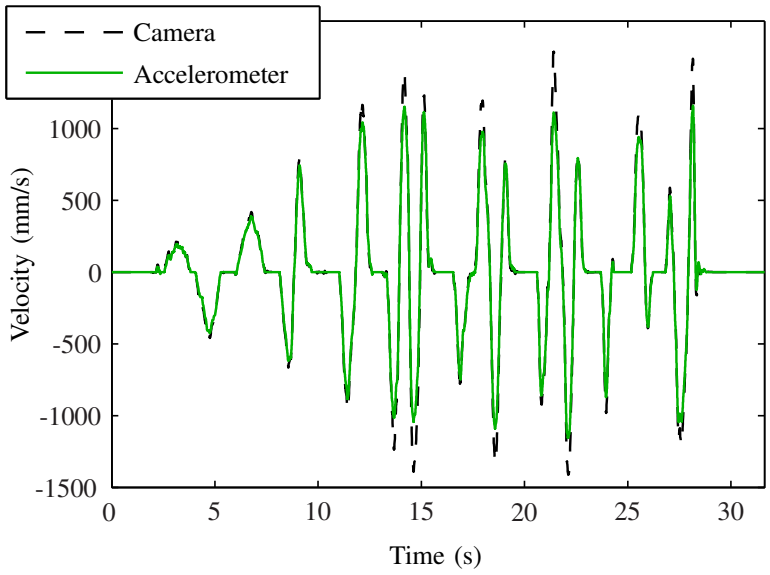


Figure 5.10: Velocity estimate derived from the accelerometer using postprocessing techniques compared to the reference provided by the Vicon vision system.

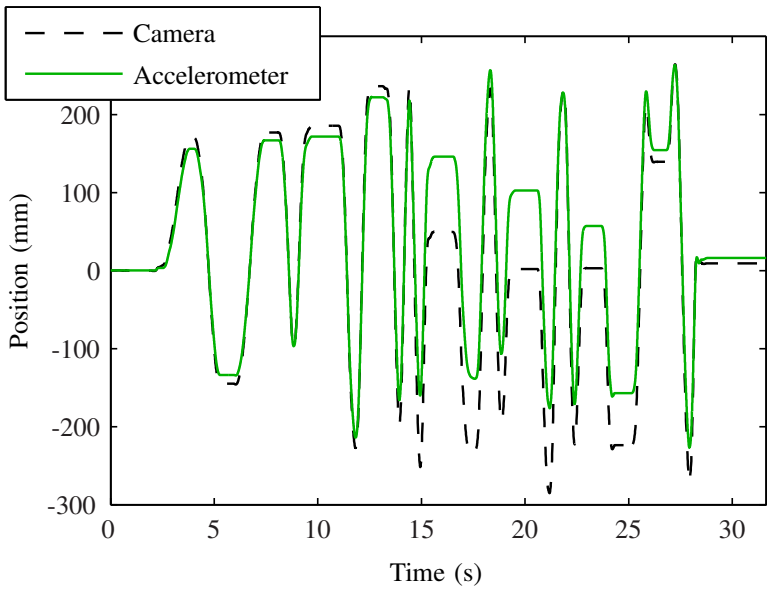


Figure 5.11: Position estimate derived from the accelerometer using postprocessing techniques compared to the reference provided by the Vicon vision system.

algorithm could potentially be applied include mobile robots with short cruising periods, end effectors of stationary robots, and head tracking systems for virtual reality. Even certain types of more general human motion tracking would be suitable, such as gait analysis with the inertial measurement unit located on the foot [18, 19]. In addition, movements made by the upper limbs, head, and torso are typically noncontinuous during stationary activities or those involving intermittent locomotion. Therefore, this algorithm could aid in the use of MEMS inertial sensors for applications such as monitoring posture, treating balance-related issues, and tracking a joints's range of motion outside of the confines of a motion laboratory, as is already being investigated in studies such as [21, 22].

The results of the algorithm's bias correction are essentially the same as that of a technique termed zero-velocity compensation, which is described in detail in [40]. The benefit offered by this new algorithm is its ability to automatically recognize motionless periods and act on them more quickly and without the need for a motion-control program to signal when motion has ceased. This is very important for human motion applications where such high-level knowledge of the motion is unavailable.

The algorithm was designed for real-time implementation using minimal data storage. However, in postprocessing applications, interpolating the bias level between adjustments has been shown to offer significantly increased accuracy in the velocity and position estimates. Using this technique, almost all error is eliminated from the velocity estimate even during extended periods of motion, as seen in Figure 5.10. In such cases, the position estimate can also remain useful over surprisingly long periods of time as seen in Figure 5.11.

For real-time applications, the results presented in Section 5.6 illustrate that the benefits offered by the algorithm depend strongly on the frequency of motionless periods. During periods of motion, bias corrections cannot be made and the velocity and position errors grow linearly and quadratically with time, respectively. Resulting drift in the position estimate is never corrected, but it is halted during motionless periods and its growth rate is reset to zero. Since velocity error is eliminated during pauses, the velocity estimate can remain useful through arbitrarily long periods of time by ensuring that pauses occur with sufficient frequency to prevent the error from becoming too large for the estimate to be of value. Once the typical velocity bias drift rate has been established, the necessary frequency of pauses needed to avoid exceeding a specified error in the velocity estimate can be determined.

As mentioned in Section 5.3.1, the work presented here was limited to one-dimensional motion in order to focus on the development of a sensor-level algorithm. This is because in three-dimensional motion, the identification of the bias level for accelerometers is complicated by the need to compensate for gravitational acceleration, which is dependent on orientation. In this case, it is impossible to differentiate between perceived changes in the bias level due to actual bias drift and due to errors in gravity compensation resulting from errors in the orientation estimate. However, the algorithm developed here can be applied without any modification to the output of gyroscopes since their output is nominally unaffected by gravitational fields. Therefore, this algorithm is immediately applicable in gyroscope-based orientation tracking and would largely eliminate problems with jitter

and stationary drift, as described in [17]. Furthermore, the resulting improvements in the orientation estimate would translate to improved gravity compensation for accelerometers and improve the accuracy of their derived velocity and position estimates.

5.7.2 Additional considerations related to the accelerometer

Two interesting issues were observed in the data obtained from the accelerometer:

1. The highest peaks and lowest valleys of the velocity were never captured accurately through integration of the accelerometer data even though the sensor's limits of $\pm 1.7g$ were never exceeded in the experiments.
2. During periods of motion, the velocity error always drifted upward regardless of the care taken to find the initial bias of the accelerometer while stationary.

It is believed that these factors are the result of low-level phenomena related to the operation of the MEMS accelerometer. Modeling methods such as those presented in [46] may be able to help correct these systematic errors by taking cross-axis and other similar effects into account.

5.8 Conclusion

The state estimation algorithm presented here greatly increases the accuracy and usefulness of velocity and position estimates obtained through inertial measurement for one-dimensional, intermittent motion. These results are obtained through a sensor-level algorithm that relies only on easily measured characteristics of the sensor noise without any need for application-specific parameters. If motionless periods are sufficiently frequent, accurate estimates may be obtained for many times longer than what is possible without the algorithm. This is especially true for the velocity estimate, which can remain useful indefinitely since errors can be eliminated at each pause in the motion.

Future work will include further study of MEMS accelerometers and gyroscopes to better characterize their output noise and low-level, systematic sources of error. The state estimation algorithm can then be enhanced to include these effects and further reduce

measurement errors at the sensor level. The possibility of estimating the bias level during extended periods of motion will also be investigated. It is believed that combining such sensor-level improvements with more advanced system-level estimation techniques, like the extended Kalman filter, will allow significant improvements in the accuracy and utility of MEMS-based inertial measurement units.

5.9 Acknowledgment

The authors would like to thank Dr. Mikhail Skliar of the University of Utah for his consultation regarding the filtering algorithm and the Skeletal Muscle Exercise Research Facility for providing access to their Motion Analysis Core Facility.

CHAPTER 6

THREE-DIMENSIONAL EXTENSION OF THE BIAS ESTIMATION ALGORITHM

Eric Allen Johnson, Mark A. Minor, and Stacy J. Morris Bamberg

Department of Mechanical Engineering, University of Utah

Salt Lake City, UT 84112

The contents of this chapter have been submitted to *IEEE Transactions on Robotics* for review. Formatting and other minor modifications have been applied for consistency with the remainder of this dissertation.

6.1 Abstract

A state estimation algorithm for rejecting noise and tracking bias in inertial measurement applications with intermittent motion was presented for one-dimensional motion in Chapter 5. This chapter extends that work to three dimensions to track a mobile robot's motion using a custom MEMS inertial measurement unit. The algorithm uses characteristics of the sensor noise to automatically recognize motionless periods and update the sensors' bias levels without any dependency on application-specific parameters or a high-level system model. The algorithm is applied to update the bias levels of the accelerometer and gyro triads, and to correct pitch and roll errors such that position drift due to incorrect gravity cancellation is greatly reduced. Using a camera-based motion analysis system for validation, centimeter-level positioning errors were achievable using unaided inertial measurement for trials with time spans of up to 30 s on ramps and level ground given reasonably frequent pauses in the robot's motion. The results demonstrate that MEMS inertial sensors can provide accurate motion tracking information over periods of time long enough to be useful in many applications.

6.2 Introduction

Inertial sensors based on microelectromechanical systems (MEMS) have steadily improved in performance and cost over the past two decades [6–9]. The drastic size, weight, and cost reductions they offer relative to traditional inertial measurement systems [1] has motivated researchers to evaluate their performance in a variety of applications involving motion tracking on significantly smaller scales than traditional aerospace applications. MEMS inertial sensors can provide high-bandwidth, high-resolution, and very detailed motion estimates. However, the utility of the motion estimates obtained from unaided MEMS inertial sensors has generally been very limited. This is due to their greatly reduced performance compared to tactical- and navigation-grade sensors, which leads to rapid and unbounded error growth.

In previous tests using unaided MEMS inertial sensors for motion tracking in mobile robotics and similar applications, error growth rates have allowed useful position estimates to be obtained for only a few seconds at a time [15, 42, 54, 57]. Thus, obtaining accurate motion tracking results using MEMS inertial sensors alone has generally been regarded as impossible [97]. Instead, they are typically used in combination with odometry [43, 98–100], GPS [44, 101], vision-based motion estimation [102, 103], active beacons [104], or other methods of motion tracking that offer better long-term stability. While these other methods generally offer lower resolution, lower bandwidth, or incomplete motion information compared to inertial measurement, the strengths of both sensing modalities can be leveraged using appropriate data fusion techniques.

Despite these advantages from combining inertial measurement with other types of sensors, there are many applications where using unaided inertial measurement is attractive due to factors like environmental considerations, size, cost, and the complexity of incorporating additional sensor types. An inertial measurement unit (IMU) can potentially provide angular rate and orientation data along with acceleration, velocity, and position estimates all from a single, self-contained source that operates independently of any outside reference. It also offers excellent flexibility since, ideally, the motion estimates do not require application-specific models or processing methods. As the size and cost of MEMS sensors continues to decrease and the level of integration offered continues to improve [48], there is ever more motivation to improve the accuracy of their outputs by developing methods to mitigate present obstacles such as parameter drift and accurate calibration.

The work presented in Chapters 3 and 4 has provided an inexpensive method of calibrating MEMS inertial sensors using sensor models that account for scale factor nonlinearity, anisotropic effects, and gyro specific-force sensitivity. This was found to reduce the calibrated sensor output error by up to 7.4% for the tested accelerometers and 70.2% for the gyros relative to standard linear sensor models. However, since velocity, position, and orientation estimates are obtained through integration, they are very sensitive to any

remaining sources of error. As a first step toward reducing these errors, the state estimation algorithm presented in Chapter 5 was developed. It was demonstrated to greatly increase the accuracy and utility of velocity and position estimates obtained for one-dimensional, intermittent motion in laboratory experiments by tracking and compensating for drift in an accelerometer's output. This was accomplished using a sensor-level algorithm that relies only on easily measured characteristics of the sensors' output noise without any application-specific thresholds or heuristics.

This chapter presents an extension of this state estimation algorithm to three dimensions and demonstrates its application to a mobile robot. Experiments were performed on both uneven and level ground using a custom-built MEMS IMU and a mobile robot designed for inertial measurement experiments. Validation of the IMU's motion was provided by a camera-based motion analysis system. The state estimation algorithm uses the same sensor noise characteristics as the one-dimensional algorithm in Chapter 5 to identify periods where both sensor output drift and accumulated pitch and roll errors can be corrected. As in the one-dimensional case, the algorithm does not rely on any application-specific assumptions and only requires that the motion be intermittent, as with other methods that utilize zero-velocity updates (ZUPTs), or zero-velocity detection (ZVD) [18,35,38,40,41].

Results are shown using discrete and interpolated bias updates, similar to the two processing methods shown in Chapter 5. Using discrete bias updates, which require very little processing power to implement, position errors of 34–76 mm, or 2.1–3.5% of the path length, and heading errors of 0.34–1.6° were obtained on both level and uneven ground for trials lasting 14–30 s with average motion segment durations of 1.5–2.5 s and maximum motion durations of 3.0–5.5 s. Using interpolated bias updates, which improve accuracy, but are more computationally intensive, position errors were reduced to 7–39 mm, or 0.35–1.8% of the path length. For a trial on level ground with an average motion segment duration of 4.5 s and maximum motion duration of 6.6 s, position errors increased to 9.8% of the path length for discrete bias updates and 2.0% for interpolated updates. It

is emphasized that these results were all obtained using a completely unaided MEMS IMU and processing methods that are devoid of application-specific assumptions or parameters.

This represents a significant improvement relative to prior attempts at motion tracking for mobile robots and other similar applications using unaided inertial measurement. In a study using early solid-state inertial sensors [15], position error was found to accumulate at a rate of approximately 280 mm/s, which could render the estimates useless in under 1 s. More recent tests using unaided MEMS inertial sensors presented in [42] were also hampered by similar rates of error accumulation. In contrast, our results have error levels comparable to those obtained in the very successful pedestrian tracking studies presented in [18, 35]. However, our method employs inexpensive, commercial-grade MEMS sensors; does not require the use of magnetometers, as in [18]; and achieves good results with motion segments lasting more than ten times longer than the typical 0.4–0.5 s per step in walking.

The level of accuracy obtained is comparable to results using a variety of other relative motion tracking methods, including wheel odometry, visual odometry, and combinations of inertial measurement with odometry using sensor fusion techniques. A table of representative results from the literature is provided in Section 6.7 and shows that maximum position errors for these different methods are typically in the range of 0.25–1.5% of path length. For experiments with a total duration on the order of 30 s, such as those demonstrated here, the position tracking errors from the IMU are also competitive with several absolute position tracking techniques typically used in mobile robotics, including differential GPS, ultrasonic ranging, and ultra-wide-band RF tracking. These results show that the state estimation algorithm presented here allows accurate, three-dimensional motion estimates to be obtained using commercial-grade MEMS inertial sensors in mobile robotics and other applications with intermittent motion if the duration of individual motion segments does not exceed approximately 5–6 s.

The remainder of this chapter is organized as follows. First, practical considerations regarding the key sources of error in three-dimensional inertial measurement are presented

in Section 6.3. The conclusions from this section guide the extension of the original one-dimensional state estimation algorithm to spatial motion tracking in Section 6.4. The experimental setup and procedures are described in Section 6.5, including a brief description of the calibration technique and the numerical integration methods used to process the data. This section also presents a simple and effective means of aligning the IMU's internal coordinate system to the reference frame provided by the vision system. Experimental results demonstrating the levels of accuracy that can be obtained for robot motion tracking are presented in Section 6.6. Discussions of the accuracy of the orientation and position estimates, along with a comparison to other motion tracking techniques, remaining limitations, and potential applications of this method are presented in Section 6.7. Finally, concluding remarks are found in Section 6.8.

6.3 Relative contributions of error sources

The design of the three-dimensional state estimation algorithm presented in Section 6.4 is motivated by an understanding of the relative contributions of the primary sources of error in motion tracking using MEMS inertial sensors. For current levels of performance of commercial-grade sensors, there are two main contributors to position and velocity errors:

- Inaccuracy in the accelerometer outputs due to calibration errors, parameter drift, output noise, and other inherent limitations in the accuracy of the sensor's outputs.
- Imperfect gravity cancellation due to errors in the orientation estimates derived from the gyros. These are due to inaccuracies in the gyro outputs that are analogous to those for the accelerometers.

In addition to these sources of error, numerical inaccuracies also result from discretizing the sensor outputs and using approximate numerical integration methods. However, given adequate sampling frequencies and resolutions, these types of errors are negligible relative to the two primary sources of error listed above [1, 3, 37]. In the remainder of this section, their relative contributions to position and velocity errors will be compared via a theoretical analysis and experimental validation.

6.3.1 Theoretical comparison

For a tilt angle error of $\delta\theta$, the error in the cancellation of gravity will create acceleration measurement errors in the horizontal direction of:

$$\delta a_h = g \sin \delta\theta \tag{6.1}$$

and, in the vertical direction:

$$\delta a_v = g\,(1 - \cos \delta\theta) \tag{6.2}$$

where g is the magnitude of the gravity vector. For small angles, $\delta a_v \approx 0$, and thus $\delta a_h \approx g\,\delta\theta$ dominates the resulting kinematic acceleration error. If the average measurement error from the accelerometer is δa_a, then the contribution to the total acceleration error due to δa_h will exceed δa_a when $\delta\theta > \delta a_a / g$.

As discussed in Section 6.5.2, the calibration process used in this study provides estimates for the standard deviation of the output error of the IMU's accelerometers and gyros. These can be used as simple, yet reasonable approximations to the average accelerometer and gyro measurement errors. Doing this for the accelerometers gives $\delta a_a = 19.7$ mm/s^2. The tilt error that would cause gravity cancellation errors to exceed this is only 0.115°. Given that the standard deviation of the gyro output error is estimated to be 0.0945°/s, that angle will be reached after only 1.2 s. This is illustrated graphically in Figure 6.1, where the acceleration, velocity, and position errors corresponding to these sensor output error values are plotted. It is assumed that the gyro output error is causing tilt errors to grow at that rate about a horizontal axis and that the accelerometer output error is constant and directed along a perpendicular axis.

The results in Figure 6.1 show that for these values of accelerometer and gyro output errors, orientation error becomes the larger source of: acceleration error after 1.2 s, of

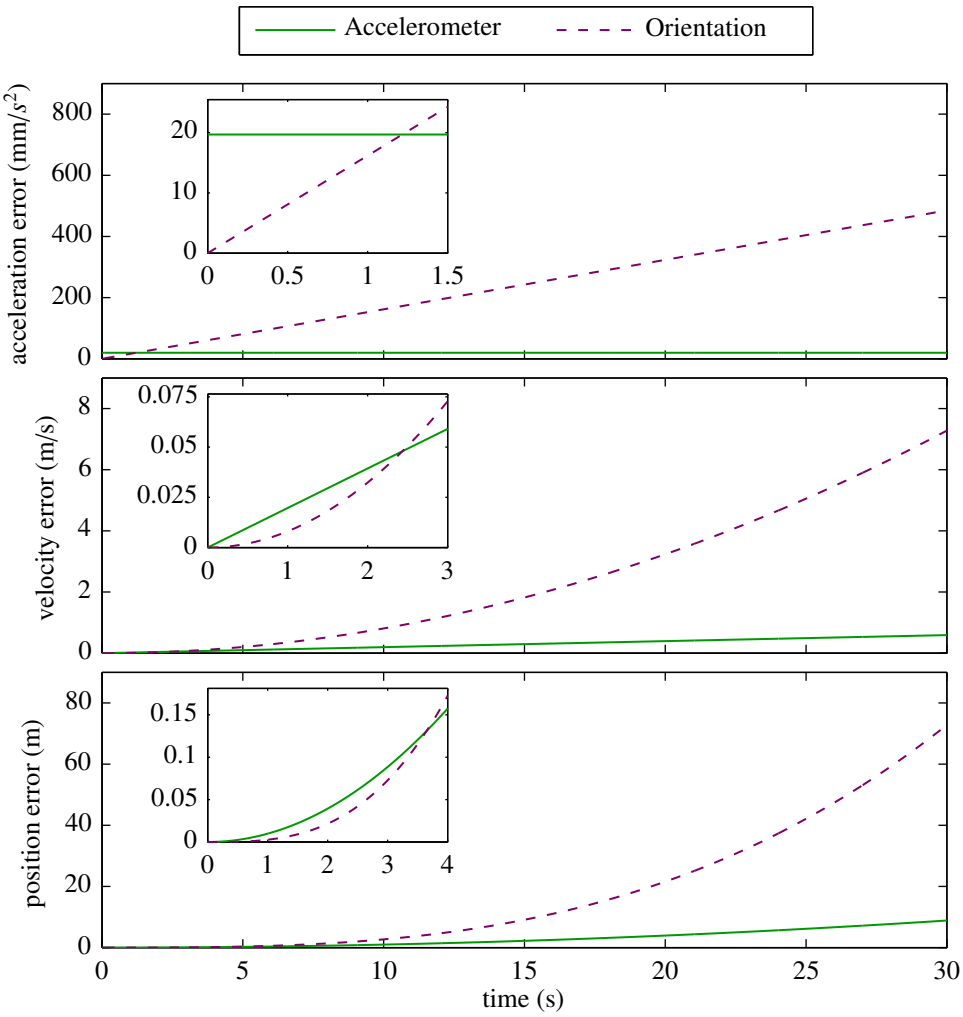


Figure 6.1: Theoretical comparison of the motion tracking errors due to a constant accelerometer output error compared to a tilt angle orientation error that is growing linearly with time due to gyro output error. The simulated output errors are the estimated standard deviations for the IMU's sensors after calibration.

velocity error after approximately 2.4 s, and of position error after approximately 3.5 s. Over the 30 s duration shown in the plot, the position error resulting from the orientation error exceeds that from accelerometer output errors by nearly ten times. This is consistent with the simulations and experiments using a commercial MEMS IMU presented in [37], where the author also concludes that orientation error is the dominant source of positioning error for integration of MEMS IMU outputs for durations beyond a few seconds.

6.3.2 Experimental Validation

The following experiment was performed to verify that the conclusions from this theoretical analysis would apply to the IMU used in this study in an actual trial. The sensor outputs from the first trial of the robot traversing the ramps, as described in Section 6.5.7, were processed using the IMU's calibrated parameters and integrated using the method described in Section 6.5.6 assuming the calibrated outputs were ideal. The results are shown in Figure 6.2. The right-most column of plots shows the total orientation error, the magnitude of the velocity error, and the magnitude of the position error. The orientation error is calculated by finding the rotation from the IMU's orientation estimate to the reference orientation provided by the camera. This rotation is converted into an angle-axis representation and the rotation angle is used as the orientation error. The velocity and position errors are calculated as vector magnitudes. The vertical scales of the velocity and position error plots in Figure 6.2 are matched to those of Figure 6.1 to allow direct comparison of the values.

The linear growth rate of the orientation error is clearly visible in Figure 6.2. At the end of the trial, which lasted just over 30 s, the orientation error is less than 2.5°. This gives an effective error growth rate of 0.083°/s, which is in close agreement to the value of 0.0945°/s used in the theoretical analysis based on the calibration results. The velocity and position errors also grow almost identically to the predictions from the simulated results shown in Figure 6.1. The approximately quadratic velocity error and cubic position error growth rates suggest that orientation error growth and the subsequent gravity cancellation errors are the dominant source of velocity and position error, as was expected.

To confirm this conclusion, the accelerometer data was re-integrated using the orientation history provided by the cameras to resolve the measurements into the reference frame. The result of this process is shown in Figure 6.3, where the vertical scales of the error plots are again matched to those of Figure 6.1 to allow a direct comparison. In these plots, the velocity error growth rate is now linear and the position error growth is quadratic. The

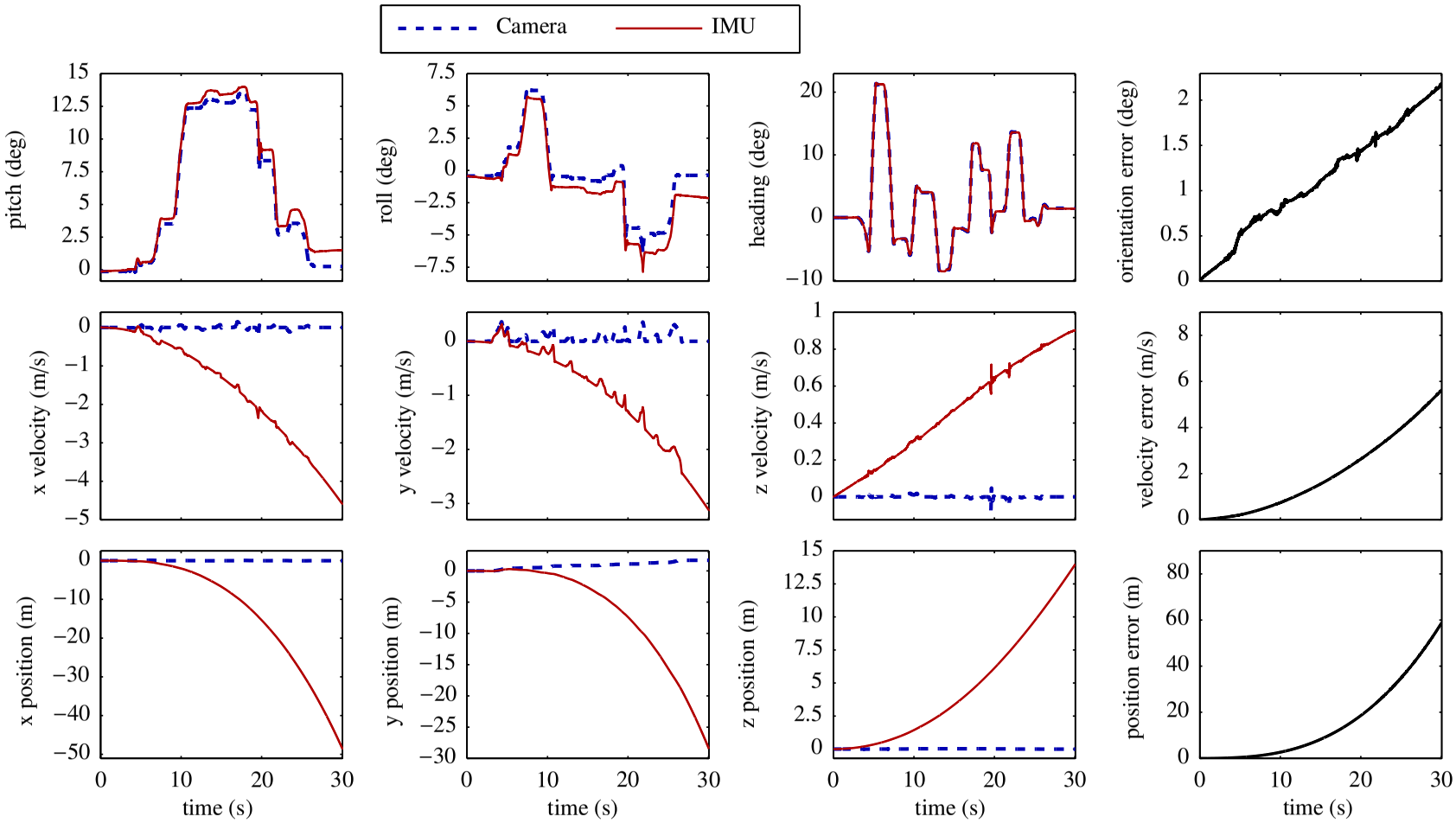


Figure 6.2: Results obtained using the calibrated sensor parameters without any bias updates during an experiment with the robot traversing the ramps.

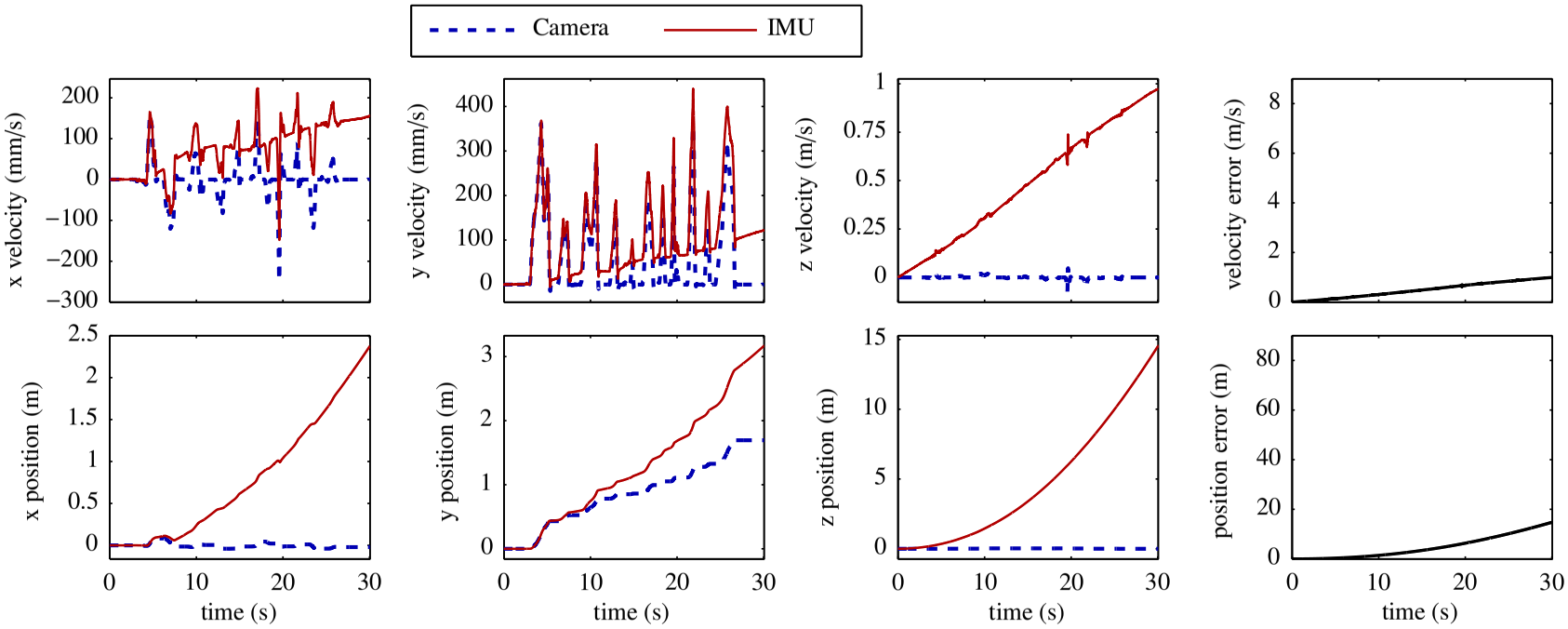


Figure 6.3: Results obtained by integrating the accelerometer outputs without any bias updates using the orientation information provided from the vision system to resolve the specific force measurements into the reference frame.

magnitudes of the final errors are again in excellent agreement with those predicted in Figure 6.1.

These results confirm that for integration periods longer than a few seconds, as would be required for most practical applications, orientation error and the resulting errors in gravity cancellation are the dominant source of position error. Because of this, one of the primary objectives of the bias tracking algorithm described in the following section is to reduce gravity cancellation errors as much as possible.

6.4 Three-dimensional extension of the bias tracking algorithm

This section presents a method of extending the state estimation algorithm from Chapter 5 for use in three-dimensional applications. The method allows automatic identification of periods where acceleration and angular rate inputs are sufficiently close to zero to not be accurately measurable by the sensors. In most applications, this will correspond to periods where motion has stopped; therefore, for convenience, these will be referred to as still periods. As in the one-dimensional case, recognition of still periods is done using easily identified characteristics of the sensor noise that can be derived from static measurements. This section defines the quantities that should be monitored and the processing steps necessary to allow the principles from the one-dimensional algorithm to be used effectively in three-dimensional motion tracking. The details of the implementation for each quantity being monitored are provided in Chapter 5.

It should be noted that drift in the sensor outputs can be due to random or unmodeled systematic changes in the scale factor, bias level, or any of the other parameters in the sensor model. However, manufacturer datasheets and studies that have quantified the stability of the bias level and scale factor of MEMS inertial sensors indicate that drift in the bias level will dominate changes in the scale factor [54,58,105]. Therefore, it is assumed that all drift in a sensor's output is due to changes in the bias level.

6.4.1 Application to gyros

If the gyro outputs were unaffected by orientation, the state estimation algorithm could be applied directly to their raw outputs, as mentioned in Chapter 5. However, the calibration performed in Chapter 4 showed that the gyros have a small, but significant specific-force sensitivity. Therefore, the bias tracking should be performed in the domain of the angular rate outputs from the sensor model after applying the calibration coefficients to compensate for the specific-force dependency.

Doing this requires only a slight adjustment to the algorithm. Rather than characterizing the noise and drift characteristics of the raw voltages outputs, the calibrated outputs are characterized instead. For these experiments, the data from the static observations used in the calibration procedure were used to characterize the necessary parameters including the peak-to-peak noise and the maximum bias drift. After compensating for the specific-force dependency, it was found that the bias was sufficiently stable to simply define a maximum range of drift rather than a maximum drift rate. This maximum range is then used to help determine when the signal is within the appropriate limits to update the bias, as described in Chapter 5.

6.4.2 Application to accelerometers

Applying the algorithm to the accelerometers is significantly more complicated, since it is impossible to differentiate between orientation errors and drift in the accelerometer outputs without any external source of information. That is why an understanding of the relative contribution of incorrect gravity cancellation and accelerometer output errors is critical. Given that positioning errors due to inaccurate gravity cancellation are dominant for periods of integration beyond a few seconds, the following method was used to update the accelerometer bias during still periods:

1. Find the direction of the measured specific force vector.
2. Scale this vector to match the magnitude of gravity.
3. Take the difference between the scaled and sensed gravity vectors.

4. Use the components of this difference as the bias updates for the individual axes of the accelerometer triad.

This ensures that the magnitude of the specific force measured by the accelerometer triad will match that of gravity. As with the gyros, it corrects the bias at the level of the calibrated sensor outputs rather than the raw voltage.

To determine the periods where bias updates can be made, the magnitude of the specific force output is first monitored. This provides a simple scalar quantity that can be characterized in terms of its noise and drift, as for the outputs of the individual gyro axes. Ensuring that the specific force magnitude is within its noise band is a necessary, but not a sufficient condition, to ensure that the body is not accelerating. Therefore, once this first criterion has been met, the individual outputs from the accelerometer axes are also checked to ensure that they have been within their peak-to-peak noise limits for a sufficient amount of time to obtain accurate estimates of their average values. Following this, the period is known to have no measurable acceleration given the sensors employed, and bias updating can begin.

6.4.3 Combined algorithm for the IMU

To ensure that bias updates are made only during periods where accurate results can be obtained, the criteria for both the accelerometers and gyros were required to be met before performing updates. Given this, the combined requirements for updating the gyro and accelerometer biases are:

1. Monitor the specific force magnitude for periods that are within its noise band.
2. When this is true, check each of the accelerometer axes.
3. If all are within their noise bands, check each of the gyro axes.
4. If all are within their noise bands, the IMU is not undergoing rotation or acceleration detectable by its sensors.

Once all of these criteria have been met, integration is immediately halted since it is recognized that no measurable input is present. In addition, the following updates to the

biases of the sensor outputs, orientation, and velocity estimates are made using information about the dominant frequency of the gyro and accelerometer outputs' noise to determine how long to wait before applying the updates for each signal, as described in Chapter 5.

1. Update the gyro biases by averaging.
2. Update the accelerometer biases by averaging and using the scaling technique described in Section 6.4.2.
3. Convert the orientation representation to pitch, roll, heading Euler angles.
4. Update the pitch and roll angles using the tilt angles detected by the accelerometers.
5. If any of the linear velocity components in the reference frame are within their noise band, determined using the method described in Chapter 5, also correct their bias to return their values to zero.

Updating of the various bias values continues through the duration of the still period by averaging the data available since the beginning of the period, with a limit set on the maximum length of the data history to use. When any one of the signals from the gyro and accelerometer axes exits its noise band, the bias updating is halted and integration of the motion resumes.

By applying this set of updates, the orientation error growth is halted since the gyro biases are updated to stop any further integration. Furthermore, the combination of updating the pitch and roll estimates (based on the direction of the specific force vector detected by the accelerometers) and adjusting the accelerometer biases (to force the magnitude of the output to match gravity) causes the reported kinematic acceleration to be exactly zero. This completely halts the accumulation of velocity and position error due to either orientation error or accelerometer measurement errors, and resets the rate of position error growth to zero at the beginning of the next integration period.

The cumulative effect of this algorithm is to break up the period of integration into smaller segments, reset the position error growth rate to zero at the beginning of each segment, and significantly reduce the orientation and velocity error growth rates due to bias reductions in the sensor outputs. The periods where this can safely be done are determined

by recognizing when the sensors' outputs match their noise signatures, indicating that there is no accurately measurable input present.

6.4.4 Discrete vs. interpolated bias updates

As discussed in Chapter 5, two different methods of applying the bias updates can be used. The first uses discrete, noninterpolated bias updates. This requires minimal processing such that it could be readily executed in real-time even on resource-constrained microprocessors. In this implementation, the final value of the updated bias from each still period is assumed constant until a subsequent update during the next still period. This results in discrete jumps in the bias values, but does not involve reprocessing any prior data.

For applications that either make use of the IMU data in postprocessing or that have the necessary computing power, the bias estimates can be interpolated between updates to further reduce the motion tracking errors that accumulate during periods of motion. For three-dimensional motion, this applies to the individual accelerometer and gyro axes, the pitch and roll angles, and the velocity components. The gyro bias updates should be interpolated first and then their outputs should be reintegrated over the last motion period to obtain an improved estimate of the evolution of the orientation. Next, the pitch and roll angles should be corrected using interpolation of the errors developed during the motion period to further refine the orientation estimates. Following this, the new orientation estimates can be used to resolve the specific force measurements into the reference frame. After subtracting gravity, the acceleration biases should be interpolated to allow an improved velocity estimate to be obtained. The final step is to interpolate the velocity bias and integrate to obtain improved position estimates over the motion period.

6.5 Hardware and experimental methods

6.5.1 Mobile robot

The three-wheeled mobile robot shown in Figure 6.4 was used during the motion tracking experiments. The rear wheel is a caster wheel that is free to pivot, allowing good ma-

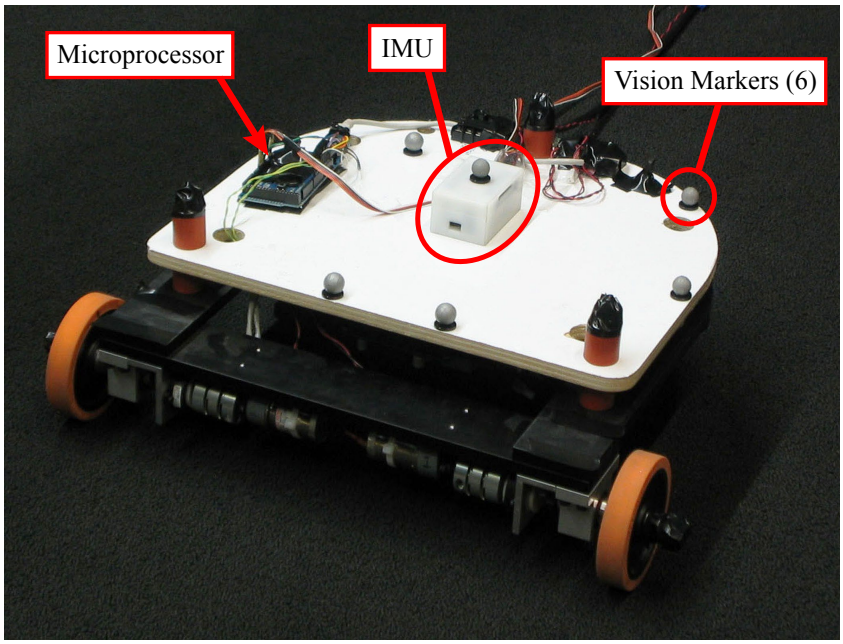


Figure 6.4: Mobile robot and IMU used in the experiments.

neuverability on level ground. The upper platform of the robot is suspended on silicon foam pads to provide vibration isolation and reduce noise in the IMU outputs during locomotion. The wheelbase of the robot is 455 mm and its mass is approximately 6 kg. During these experiments, the robot's trajectory was controlled interactively using a joystick interfaced to the microcontroller that commands the motor drivers, as shown in the system diagram in Figure 6.5.

6.5.2 Inertial measurement unit

The IMU used in these experiments can be seen mounted on the mobile robot in Figure 6.4. The external dimensions are approximately $47.5 \times 36.5 \times 74$ mm and details regarding the sensors and data acquisition system can be found in Chapter 4. Briefly, the IMU has four independent and identical accelerometer triads that provide a $\pm 2g$ output range. There are also two independent gyro triads, each of which provide two different output ranges. The first triad has ranges of $\pm 30°/s$ and $\pm 120°/s$ while the second provides

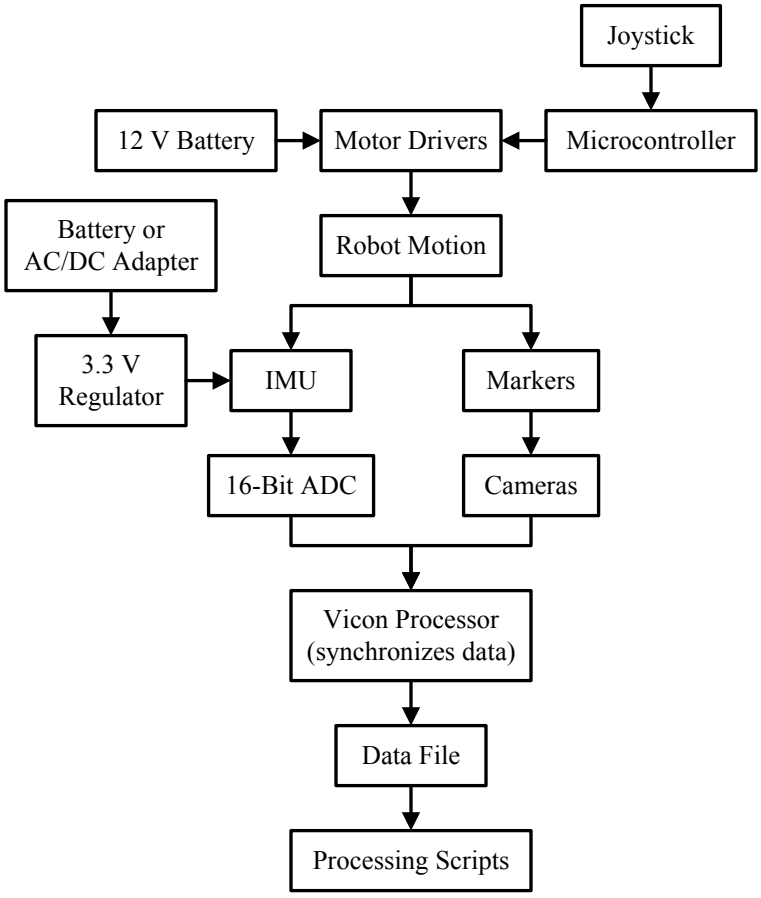


Figure 6.5: Diagram of the components and data flow in the experimental setup.

±100°/s and ±400°/s ranges. The outputs of these sensors were sampled at 1000 Hz using a 16-bit ADC with an input range of ±5 V.

In these experiments, the calibrated outputs from the separate triads were combined with weights determined according to the variances of the static outputs from the individual axes. For the gyros, only the dynamic ranges that are not saturated are used to determine the output at a given sample. The combined outputs are then used as the angular rate and specific force estimates from the IMU.

The IMU was calibrated using the technique presented in Chapters 3 and 4 immediately after performing the experiments described in Section 6.5.7 to obtain the sensor model parameters used to process the data. Given the residuals from the calibration equations, the average of the estimated standard deviations for the accelerometer triads' outputs was

19.7 mm/s^2. For the gyros, the average was 0.0945°/s. These figures are representative of the typical level of error in the IMU's outputs, and provide estimates of the attainable accuracy of the motion tracking results, as was investigated in Section 6.3.

6.5.3 Vision-based tracking system

A truth reference for the motion of the robot was provided using a Vicon MX vision-based motion analysis system. It was used to track the motion of the reflective spherical markers seen on the upper plate of the robot and the IMU in Figure 6.4. The marker centered on the top of the IMU was used as the origin of a coordinate system defined using the complete set of markers. The method of determining the alignment between this coordinate system and that of the calibrated IMU is discussed below in Section 6.5.4.

After calibration of the vision system, the software reported residual errors of approximately 0.16 mm for the marker positions. As will be seen in Section 6.6, this is more than an order of magnitude smaller than the errors in the most accurate position estimates derived from the IMU and, hence, can be considered negligible. The average distance between the origin and the other markers is 171.5 mm. Therefore, the worst-case error in the measured orientation of the marker coordinate system, or the marker frame, is expected to be on the order of $2 \cdot 0.16/171.5 = 1.87$ mrad = 0.107°.

6.5.4 Alignment of IMU and camera data

Alignment of the IMU's coordinate frame to the marker frame was accomplished by placing the robot on a swivel chair and spinning it through several rotations with the robot approximately level, on its side, and with its front facing down. The angular rate vector detected by the gyros during the spins provides an estimate of the direction of the rotation axis in the IMU coordinate frame. The rotation axis can also be estimated in the marker frame, m, using the fact that it is invariant during the rotations. If ${}^{c}\mathbf{R}_{m_0}$ is the initial orientation of the marker frame relative to the camera frame, c, and the rotation axis

expressed in the marker frame is designated as ${}^{m}\mathbf{v}$:

$$ {}^{m_i}\mathbf{R}_{m_0}\,{}^{m}\mathbf{v} = {}^{m}\mathbf{v}\ , \tag{6.3} $$

where:

$$ {}^{m_i}\mathbf{R}_{m_0} = {}^{m_i}\mathbf{R}_{c}\,{}^{c}\mathbf{R}_{m_0} = {}^{c}\mathbf{R}_{m_i}^{\mathrm{T}}\,{}^{c}\mathbf{R}_{m_0} \tag{6.4} $$

is the rotation matrix from the initial marker frame to any subsequent orientation during the rotation. Rearranging (6.3) gives:

$$ \left({}^{m_i}\mathbf{R}_{m_0} - \mathbf{I}_{3\times3}\right){}^{m}\mathbf{v} = \mathbf{0}_{3\times1}\ , \tag{6.5} $$

where $\mathbf{I}_{3\times3}$ is a 3×3 identity matrix and $\mathbf{0}_{3\times1}$ is a 3×1 vector of zeros. Stacking instances of (6.5) for multiple orientation observations during the rotations provides the following set of over-constrained equations:

$$ \left(\begin{bmatrix} {}^{m_1}\mathbf{R}_{m_0} \\ \vdots \\ {}^{m_n}\mathbf{R}_{m_0} \end{bmatrix} - \begin{bmatrix} \mathbf{I}_{3\times3} \\ \vdots \\ \mathbf{I}_{3\times3} \end{bmatrix}\right){}^{m}\mathbf{v} = \mathbf{0}_{3n\times1}\ . \tag{6.6} $$

This set of equations is reminiscent of an eigenvalue problem, which is shown to be a valid solution technique in [106]. After defining:

$$ \mathbf{A} = \begin{bmatrix} {}^{m_1}\mathbf{R}_{m_0} \\ \vdots \\ {}^{m_n}\mathbf{R}_{m_0} \end{bmatrix} - \begin{bmatrix} \mathbf{I}_{3\times3} \\ \vdots \\ \mathbf{I}_{3\times3} \end{bmatrix}, \tag{6.7} $$

then the least-squares solution of (6.6) with the constraint that $\mathbf{v}$ be a unit vector is the eigenvector of $\mathbf{A}^{\mathrm{T}}\mathbf{A}$ associated with a zero eigenvalue or, in practice, the eigenvalue with the minimum magnitude. This provides a robust estimate of ${}^{m}\mathbf{v}$ by allowing orientation information from observations throughout the rotations to be utilized.

Given the rotation axes from three separate rotations in both the marker and IMU frame, U, an estimate of the rotation matrix between them is obtained by manipulating:

$$ {}^{U}\mathbf{R}_{m}\begin{bmatrix} {}^{m}\mathbf{v}_1 & {}^{m}\mathbf{v}_2 & {}^{m}\mathbf{v}_3 \end{bmatrix} = \begin{bmatrix} {}^{U}\mathbf{v}_1 & {}^{U}\mathbf{v}_2 & {}^{U}\mathbf{v}_3 \end{bmatrix} \tag{6.8} $$

to give:

$$ {}^{U}\mathbf{R}_{m} = \begin{bmatrix} {}^{U}\mathbf{v}_1 & {}^{U}\mathbf{v}_2 & {}^{U}\mathbf{v}_3 \end{bmatrix}\begin{bmatrix} {}^{m}\mathbf{v}_1 & {}^{m}\mathbf{v}_2 & {}^{m}\mathbf{v}_3 \end{bmatrix}^{-1} . \tag{6.9} $$

Due to imperfections in the experimental data, the result of this calculation will not be a true rotation matrix. The closest orthonormal matrix can be obtained using singular value decomposition, as described in [74].

Note that this alignment approach is very similar to the traditional method of gyro-compassing and using the gravity vector to align an IMU to a local level frame, as described in [2, Ch. 6]. Here, three vectors were used to improve the robustness of the estimated rotation matrix. However, the third vector required to make the set of equations invertible could be obtained from the cross product of the first two if fewer observations are collected.

The accuracy of the alignment is demonstrated in Figure 6.6, where the angular rates in the marker frame estimated using the camera data are transformed into the IMU frame using the calibrated rotation matrix. These are compared to the angular rates reported by the gyros, which are provided directly in the IMU frame. There is no visible error between the two estimates confirming that the two coordinate systems were aligned correctly.

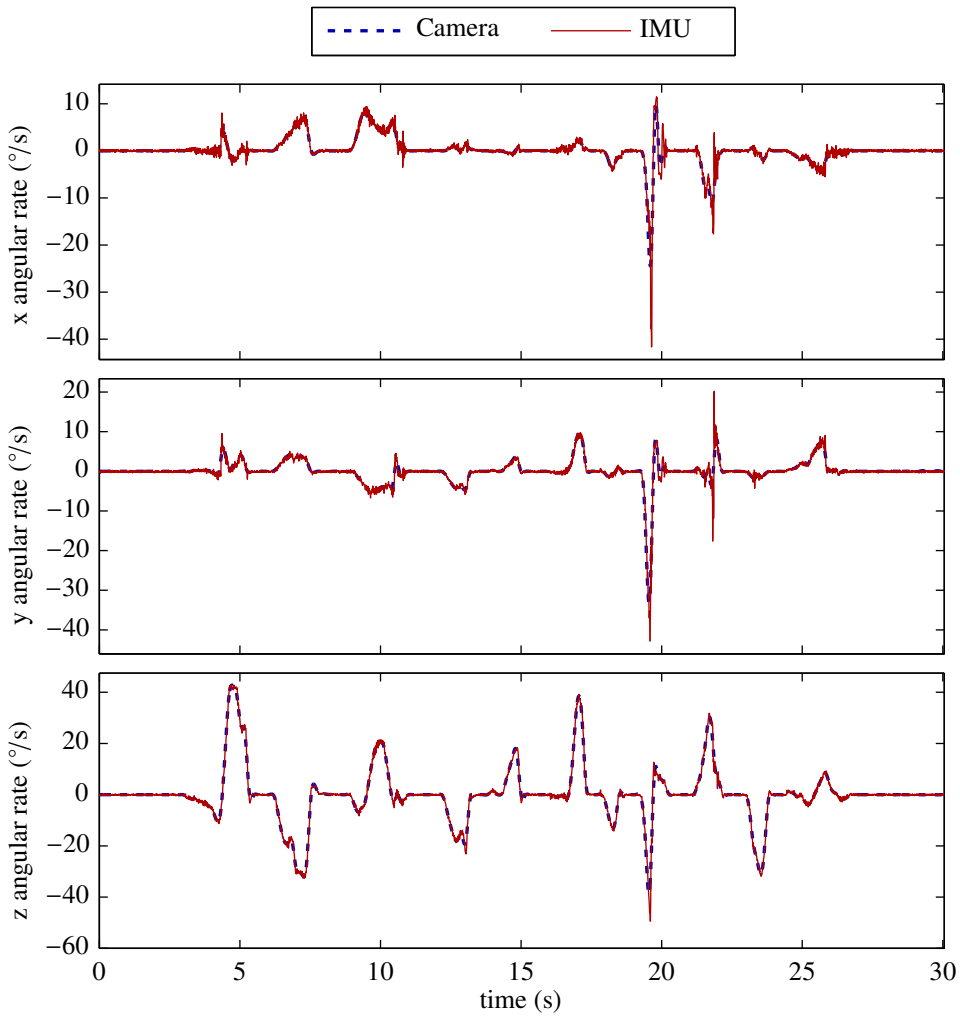


Figure 6.6: Comparison of the angular rates measured by the gyros and those derived from the camera data. Both are plotted in the frame of the IMU by using the calibrated alignment between the marker and IMU frames to transform the camera-derived data into the IMU frame.

6.5.5 Reference coordinate system definition and initialization

To simplify the data processing, the reference coordinate frame for the motion tracking experiments was defined as a local-level frame coincident with the IMU frame's origin at the start of each experiment and aligned with its initial heading. This allowed the initial position and heading angle to be taken as zero. The axes were defined to have the x axis pointing right, the y axis pointing forward, and the z axis pointing up relative to the robot and IMU. This is different than the North-East-Down definition commonly used in aviation,

but it provides a more intuitive interpretation of the motion in the z direction. Given this definition of the reference frame, pitch is measured about the x axis, roll about the y, and the heading angle is positive when the robot turns to the left.

All of the trials included an initial still period of approximately 1 s during which the direction of gravity could be estimated using the outputs from the accelerometers. This provided the initial values for the pitch and roll angles required to define the local-level reference frame. To minimize the effects of the accelerometers' output drift on the initial attitude estimate, the data from the initial still period were averaged and the direction of the resulting vector was taken as the direction of gravity for the roll and pitch angle initialization.

6.5.6 Data integration method

Integration of the IMU outputs to obtain orientation, velocity, and position data was performed as follows:

1. The defined initial heading of zero with the initial pitch and roll angles obtained from the accelerometers were converted to an equivalent quaternion representation of the initial orientation of the IMU in the reference frame.
2. The gyro outputs were integrated using the closed-form, first-order quaternion integration method presented in [107,108]. This is equivalent to a trapezoidal integration method for three-dimensional orientation and has the benefit of maintaining the unit length of the quaternion with each update. This avoids the need for frequent renormalization, which is common in other methods.
3. The orientation history provided by the quaternion integration was used to transform the accelerometer specific force outputs into the reference frame.
4. Since the reference frame is defined to be level, gravity is along its z axis. This was subtracted from the specific force outputs to obtain the kinematic acceleration in the reference frame. In doing this, the magnitude of gravity was taken to be the value

reported by the National Geodetic Survey for the location of the experiments (see Section 4.8.1).

5. The kinematic acceleration values in the reference frame were then integrated twice with respect to time using a trapezoidal, first-order approximation to obtain the velocity and position estimates in that frame.

6.5.7 Experiments conducted

Two types of experiments were conducted. First, to simulate locomotion on rough terrain, wooden ramps were placed in the configuration shown in Figure 6.7, and the robot was driven across them. The caster wheel in the rear of the robot is, obviously, not ideal for stability during such tests. But, it served well to increase the variations in the robot's posture. Four tests were conducted traversing the ramps with different frequencies of pauses. In the first test, the robot was driven very carefully over the ramps in short motions with brief pauses between them. In the last, pauses were made only as necessary to evaluate the position of the robot and avoid a premature dismount.

To provide more sustained motions, tests were also conducted with the robot driving on level ground. These tests allowed longer periods of motion between pauses to assess

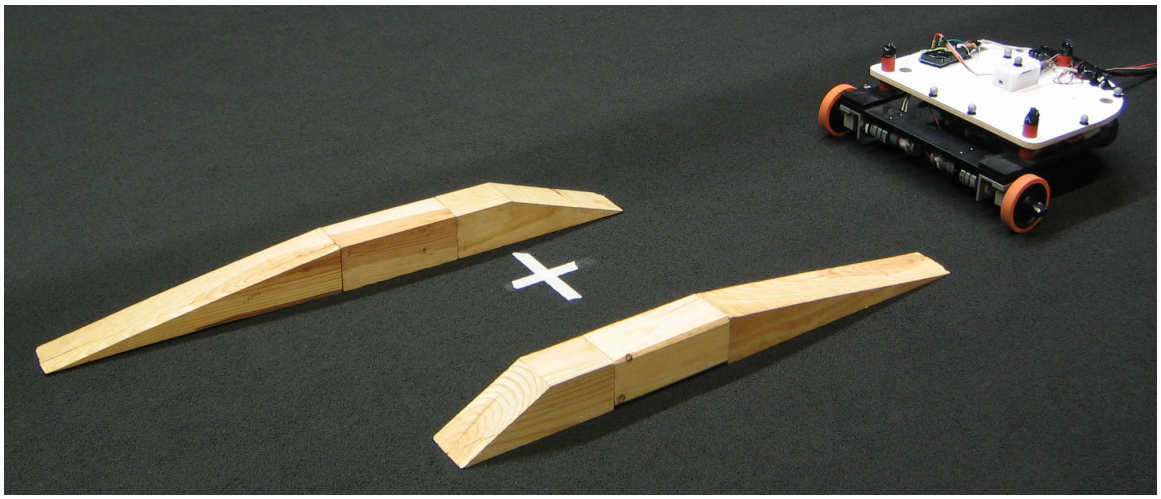


Figure 6.7: The mobile robot prepared to drive over the wooden ramps used to create pitch and roll as in rough terrain.

the results of the data processing algorithms when fewer opportunities for bias updates are present.

6.6 Experimental results

To demonstrate the state estimation algorithm's ability to identify periods where bias updating can be performed accurately, Figure 6.8 shows several seconds of data from each of the axes of the gyro and accelerometer triads. Both are resolved into the reference frame, and the accelerometer outputs have gravity removed to move the signals into a common range. During still periods, it is evident that the signals from all axes reduce to within the noise band. The dashed black boxes in Figure 6.8 indicate the periods identified by the algorithm as still period, which were then used to update the various bias levels.

The motion tracking results using both discrete and interpolated bias updates, as dis-

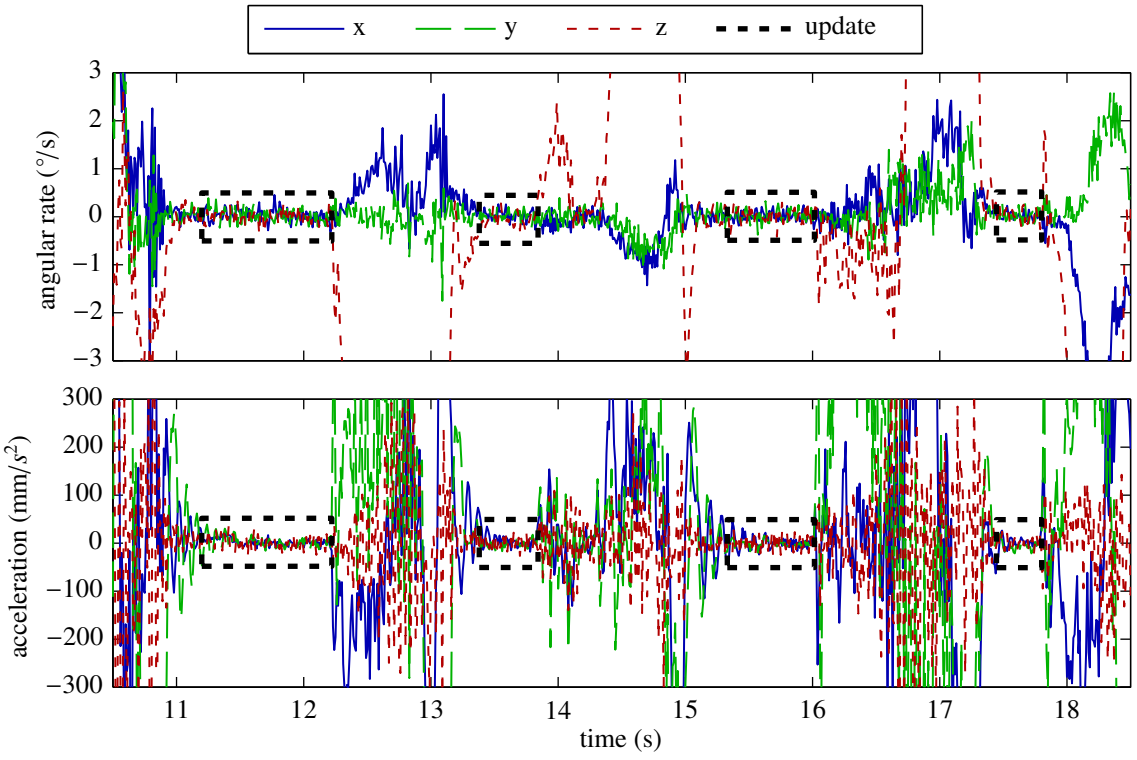


Figure 6.8: Illustration of the state estimation algorithm identifying periods where the accelerometer and gyro biases can be updated.

cussed in Section 6.4.4, are summarized for the four trials conducted on the ramps and two trials conducted on level ground in Table 6.1. Values that change depending on the type of bias update used have the results listed separately in the appropriate columns. It can be seen that using interpolated bias updates provides only modest improvements in the maximum pitch and roll errors, but the velocity and position errors are reduced significantly. Since heading angle errors cannot be corrected using inertial sensors alone, there is no difference in the maximum heading error between the two methods. That is why the corresponding column lists only a single value rather than separating the results by discrete and interpolated bias updates.

To provide a more detailed illustration of the accuracy of the motion tracking using the IMU, the results using discrete bias updates for the first trial on the ramps are shown in Figure 6.9. Note that this is the same trial that was illustrated in Figure 6.2 using no bias updates. Using the discrete bias updates, the position error is seen to remain under 40 mm throughout the 30 s trial. This is roughly six times more error than using interpolated updates. The final orientation error is under 0.1° and it reached a maximum of only 0.37°. The errors in the pitch, roll, and heading angles (determined relative to the vision system's estimates) are shown individually in Figure 6.10. The errors in the three velocity components are shown in the reference frame in Figure 6.11. The effects of the discrete bias updates are evident in the errors for the pitch and roll angles along with the velocity components. Each time a still period is detected, the error is suddenly brought to zero as the bias update is applied. The pitch and roll errors during still periods are consistently below 0.1°, which is equal to the estimated worst-case error in the orientation tracking via the vision system, as noted in Section 6.5.3.

The orientation and position tracking results using interpolated bias updates for the first trial on the ramps are shown in Figure 6.12. The vertical scales on the error plots are the same as those in Figure 6.10 to allow direct comparison. At the end of the trial, the orientation error is just over 0.2°, while the position error is, remarkably, only 6.7 mm.

Table 6.1: Summary of experimental results using discrete (Disc.) and interpolated (Int.) bias updates.

Trial Type	Trial #	Duration (s)	Time in motion (s)	Avg. motion duration (s)	Max. motion duration (s)	Max. heading error (deg)	Max. pitch + roll error (deg)		Avg. velocity error (mm/s)		Path length (mm)	Max. position error (mm)		Max. pos. error / path length	
							Disc.	Int.	Disc.	Int.		Disc.	Int.	Disc.	Int.
Ramps	1	30.1	18.5	1.54	3.05	0.34	0.27	0.25	3.0	1.8	1931	40.1	6.7	2.08%	0.35%
	2	23.8	14.6	1.82	3.02	0.74	0.28	0.27	4.4	2.6	1729	60.0	19.4	3.47%	1.12%
	3	14.0	10.2	2.54	4.47	0.74	0.36	0.33	5.6	3.5	1327	34.4	18.9	2.59%	1.42%
	4	14.6	9.3	2.33	4.53	0.48	0.50	0.46	9.5	5.7	2186	75.6	38.5	3.46%	1.76%
Level	1	27.1	19.7	2.46	5.54	1.58	0.72	0.40	17.2	7.6	6793	177.5	74.2	2.61%	1.09%
	2	29.4	22.3	4.46	6.60	1.65	1.08	0.73	62.5	20.6	12383	1207.7	248.6	9.75%	2.01%

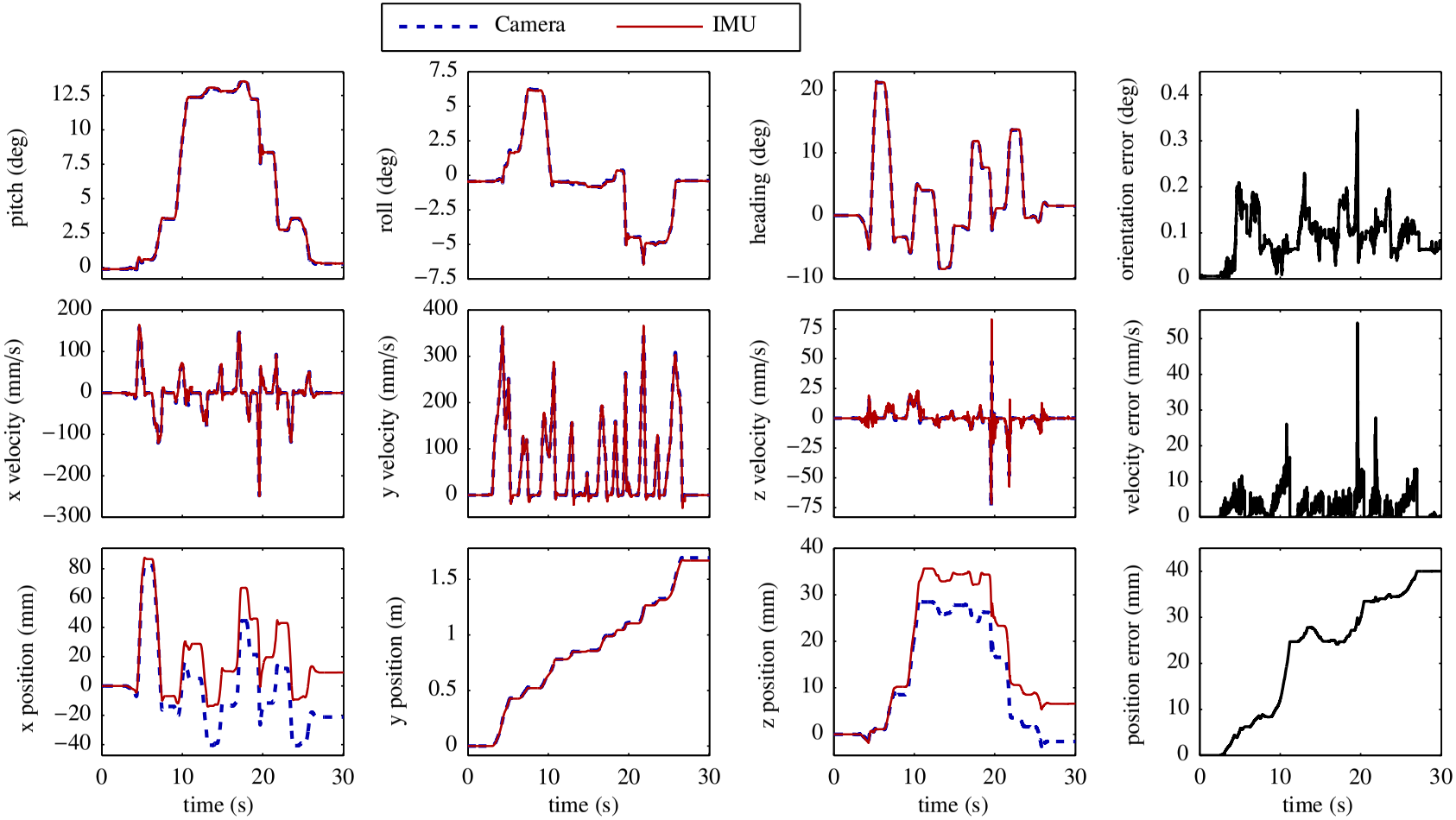


Figure 6.9: Motion tracking results obtained from the IMU for the first trial on the ramps using the state estimation algorithm with discrete bias updates.

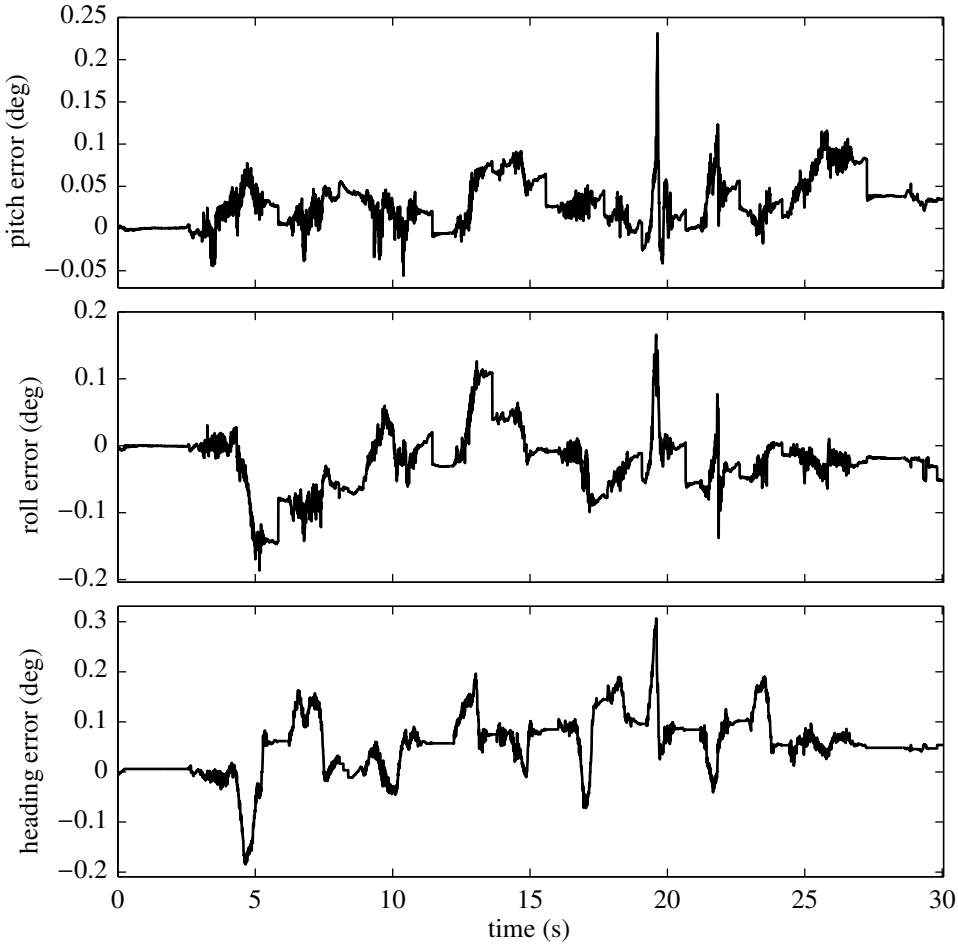


Figure 6.10: Errors in the pitch, roll, and heading angles relative to the vision system's estimates using discrete bias updates.

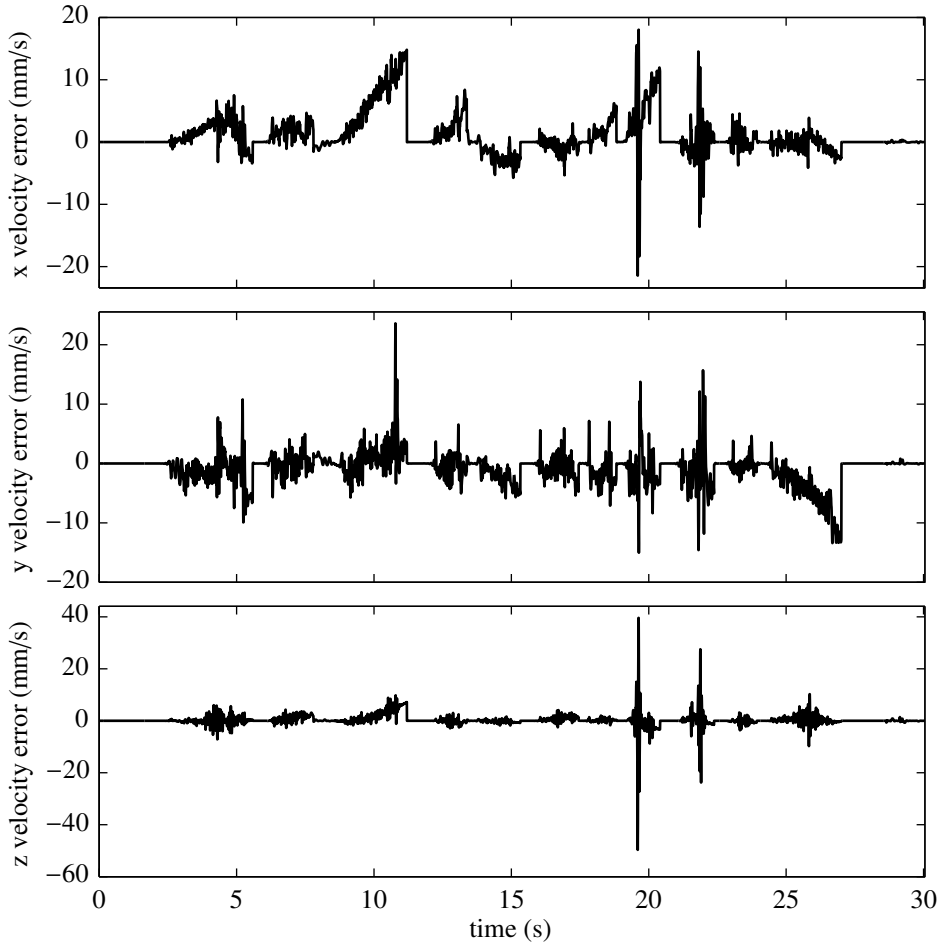


Figure 6.11: Errors in the components of velocity in the reference frame relative to the vision system's estimates using discrete bias updates.

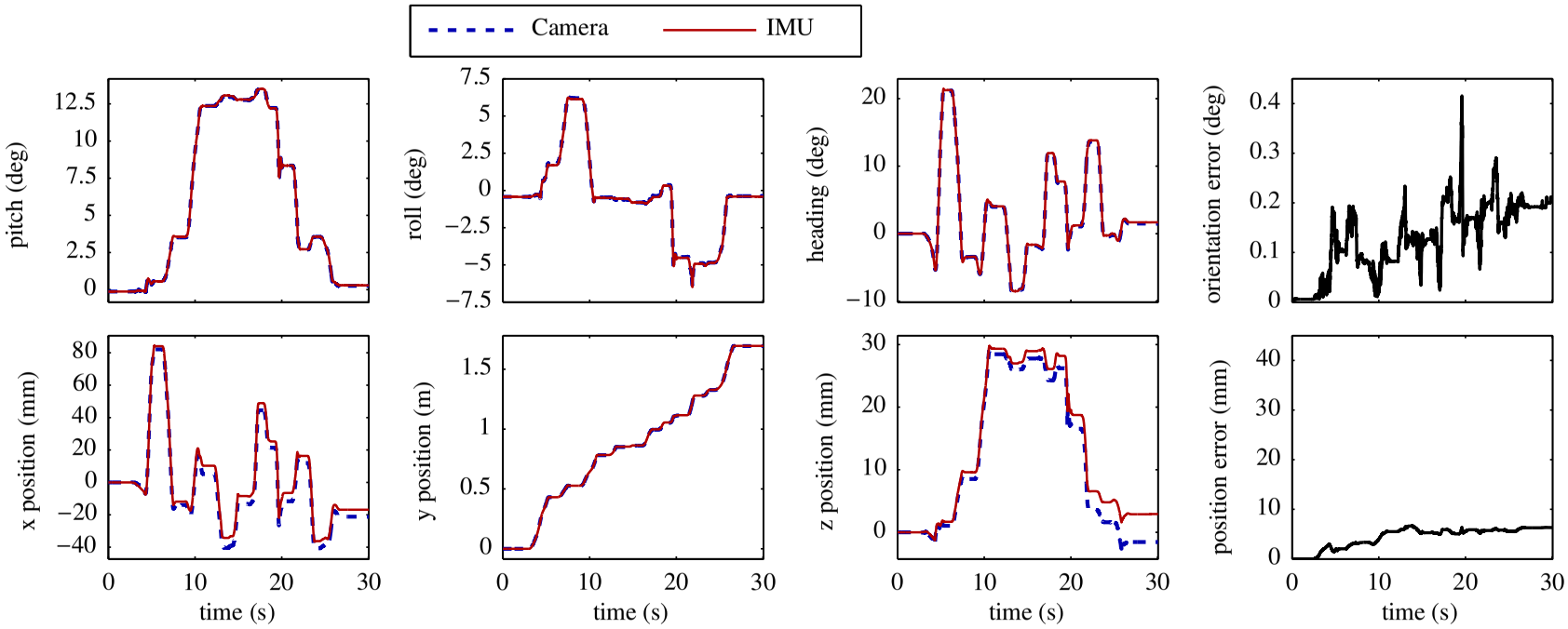


Figure 6.12: Motion tracking results obtained from the IMU for the first trial on the ramps using the bias tracking algorithm with interpolation between updates.

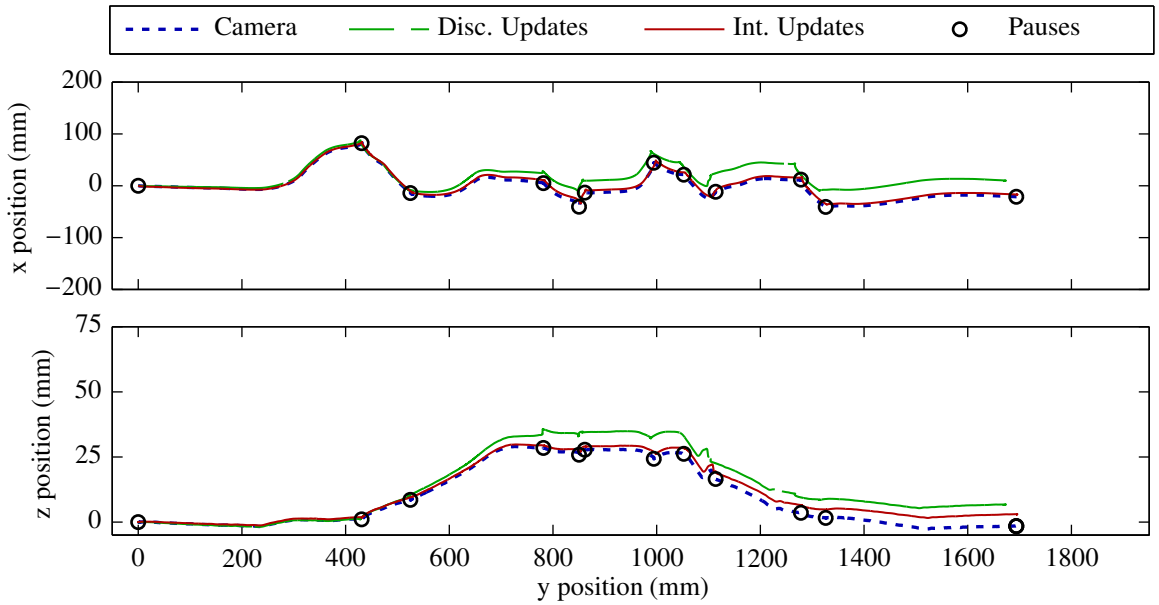
As shown in Table 6.1, the maximum period of integration between bias updates for the first trial on the ramps was 3.05 s and the average was 1.54 s. These were reasonable lengths of time between pauses given the need to frequently assess the position and stability of the robot on the ramps. The true and estimated paths traversed and the detected pauses in the motion for the first and fourth trials on the ramps are shown in Figure 6.13. The fourth trial had significantly longer periods of integration between pauses with a maximum duration of 4.53 s and an average of 2.33 s. With the increased time between bias updates, the reduced accuracy in the position estimates from the IMU relative to the first trial is evident. However, the features of the path were still captured reasonably well, especially when using interpolated bias updates.

The trials conducted on level ground provide an example of the results that can be obtained in an application with longer cruising times between still periods, as would be common for an application using a mobile robot indoors or on pavement. The motion tracking results for the second trial on level ground using discrete bias updates are shown in Figure 6.14 and the results using interpolated updates are shown in Figure 6.15. The second trial is highlighted because it provides greater distinction between the results with and without interpolated bias updates. In this trial, the maximum length of integration between still periods was 6.60 s and the average length of integration periods was 4.46 s. For both discrete and interpolated bias updates, the maximum orientation error is below 2° and the final orientation error is approximately 1°. The maximum position error drops from 1.2 m using discrete updates to under 25 cm using interpolation between updates.

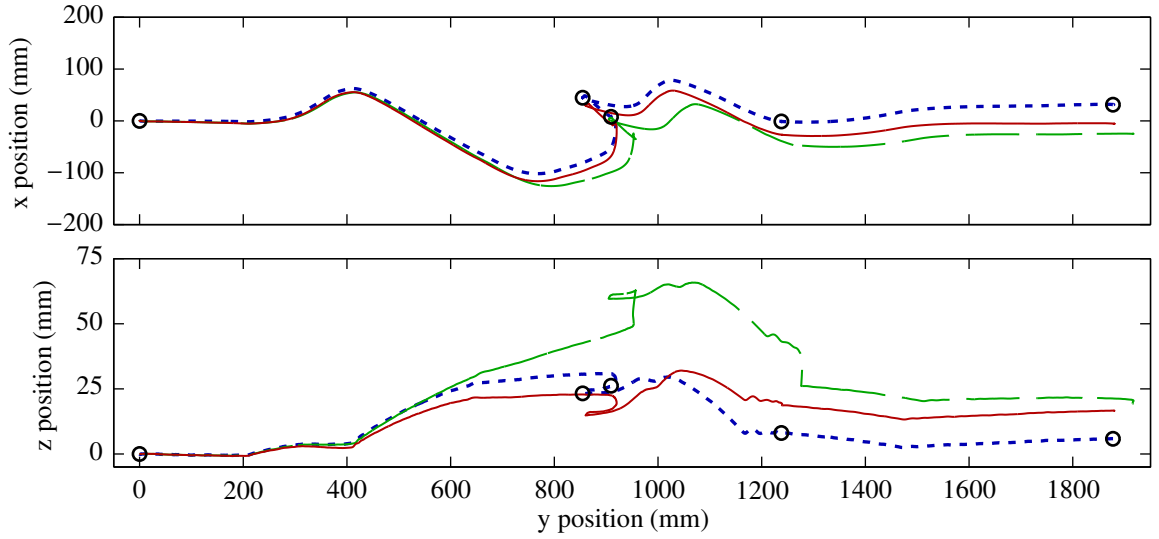
6.7 Discussion

6.7.1 Orientation tracking

The accuracy of the orientation estimates derived from the gyros alone was very good even without any bias tracking. The drift rate of approximately 0.1°/s could provide subdegree accuracy for 10 s without any aiding. For applications requiring only attitude and heading information, such as head tracking in virtual reality applications or estimating



(a) First ramps trial



(b) Fourth ramps trial

Figure 6.13: Top (x/y) and side (z/y) views of the true path traversed, as measured by the cameras, and motion tracking results from the IMU using discrete (Disc.) and interpolated (Int.) bias updates for the first (a) and fourth (b) trials traversing the ramps. The locations of the detected pauses in the motion are also indicated.

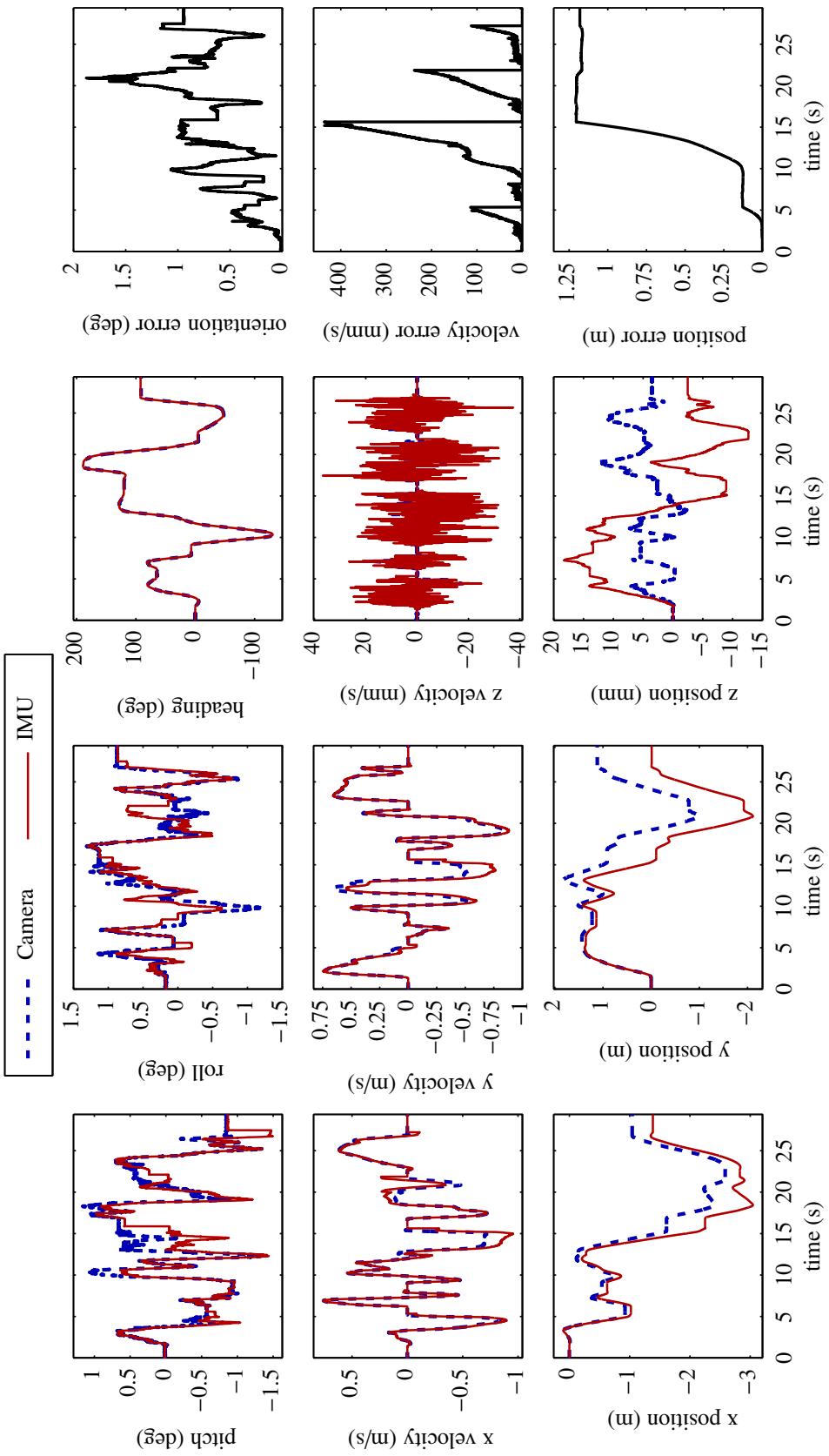


Figure 6.14: Motion tracking results obtained with discrete bias updates for the second trial with the robot driving on level ground.

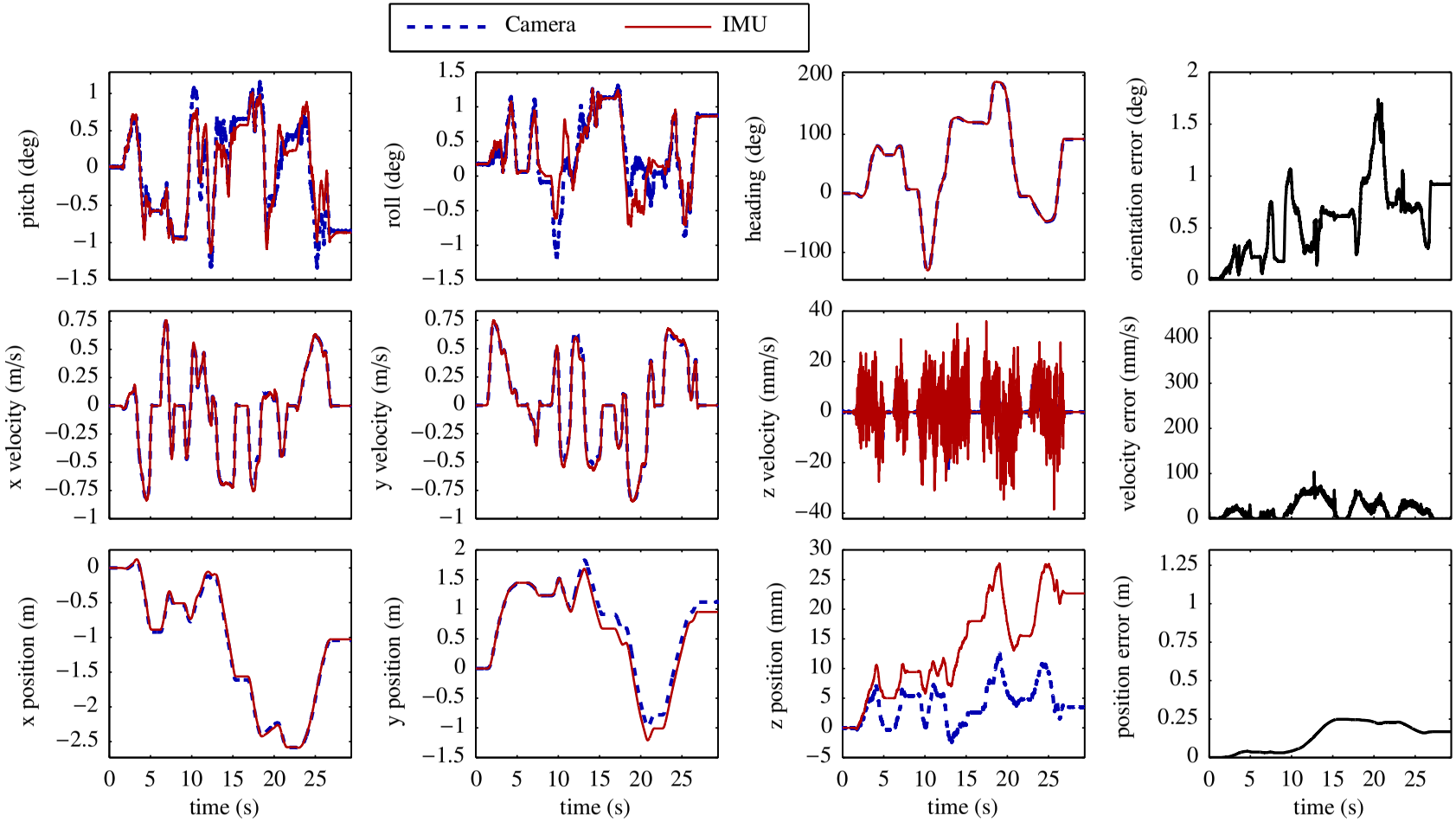


Figure 6.15: Motion tracking results obtained using bias interpolation for the second trial with the robot driving on level ground.

ground reaction forces on a mobile robot negotiating uneven terrain, these levels of error would be tolerable for short experiments. This is especially true if the motion is intermittent, allowing bias updates to be applied to the gyro outputs to halt orientation error growth during still periods.

For applications requiring position and velocity estimates, it was shown that the small amount of orientation error that can accumulate after only 1 s causes gravity cancellation errors to surpass the errors in the accelerometer outputs. The three-dimensional bias tracking algorithm described in Section 6.4 was shown to very effectively reduce gravity cancellation errors by limiting the growth of pitch and roll errors. As seen in Figure 6.10, the algorithm's use of aiding information from the accelerometers allows pitch and roll errors to be kept in the vicinity of 0.1°. This is below the point identified in Section 6.3.1 where gravity compensation errors exceed accelerometer output errors.

Interpolation of the pitch and roll errors was found to provide the greatest benefit in the trials on level ground, where there were longer periods of integration. For the first trial on the ramps, illustrated in Figure 6.9 and Figure 6.12, the maximum pitch and roll errors were reduced slightly by using interpolation, but the overall orientation error at the end of the trial actually increased. This is because only the pitch and roll angles are corrected, which can negatively impact the heading calculation due to the interaction between the three angles. Although this causes greater total orientation error, the interpolation is still desirable, because it reduces the error in the pitch and roll angles. This results in more accurate gravity cancellation, and improved total position error, as seen in Table 6.1 and Figure 6.13.

The heading angle errors do not contribute as rapidly to position error because they do not cause errors in the kinematic acceleration estimates. However, they do still cause position error to accumulate proportionally to the heading angle error and the distance traveled, just as with wheel odometry. While a major goal of this work was to demonstrate the attainable accuracy using unaided MEMS inertial sensors, the heading angle errors could be reduced by incorporating magnetometers for applications in environments free from

significant magnetic disturbances [18, 39, 105]. This can be challenging in mobile robotics applications where permanent magnet motors and large electric currents create such disturbances. As an example, in [109], the magnetometers had to be mounted on a mast that raised them approximately one half meter above the robot to avoid this type of interference.

6.7.2 Velocity and position estimation

The results demonstrated in these experiments clearly show the effectiveness of the proposed algorithm in improving the position and velocity estimates obtained using unaided inertial sensors. By reducing errors in gravity cancellation, and performing other appropriate bias updates, the velocity estimates can remain useful indefinitely given intermittent motion. Additionally, the accuracy of the position estimates were improved by four orders of magnitude relative to the results obtained by directly integrating the calibrated IMU outputs for a trial lasting 30 s.

The choice made in the state estimation algorithm to preserve the direction of the specific force vector and update the accelerometer bias accordingly is, certainly, a simplification. Parameters beyond just the bias could also have drifted and there is no guarantee that the indicated direction is correct. However, taking this action allows gravity cancellation errors to be halted, which has been shown to be paramount in reducing the growth of errors in the position estimate.

With pauses every few seconds while traversing uneven terrain, position error on the order of a few centimeters was obtained after 30 s with a path length of around 2 m. The position error grew to tens of centimeters over the same time span driving on level ground with longer periods of motion and a total path length of over 12 m. In both cases, there is a significant improvement in the position estimate by interpolating the bias levels. However, since heading errors remain uncorrected using unaided inertial sensors, they cause a significant amount of the remaining position error. This can be seen clearly in the top views of the paths in Figure 6.13. As mentioned in the previous section, these

errors could be reduced in applications where magnetometers can be used since this would improve the heading estimates.

6.7.3 Comparison to other motion tracking techniques

As discussed above, updates to the bias levels of the sensors made during static periods allow accumulated errors in the velocity, pitch, and roll to be eliminated, which allows position errors to be reduced via appropriate data processing techniques. In this way, the method is very similar in its effects to various zero-velocity compensation techniques [18, 20, 35, 40, 41, 110, 111]. However, the state estimate algorithm presented here utilizes a framework that tunes the detection of still periods to the sensors being used rather than the specific application. This eliminates the need to experimentally adjust thresholds and other heuristics for different applications, thereby providing improved flexibility and wider applicability.

To provide a comparison of the motion tracking accuracy achieved in these experiments with that of other methods currently used in mobile robotics applications, Table 6.2 provides a summary of typical results for several motion tracking methods that use an absolute position sensor and Table 6.3 provides results for a variety of relative motion tracking techniques. Comparing to the experimental results summarized in Table 6.1, it can be seen that the accuracy obtained using the unaided MEMS IMU and state estimation algorithm presented here is comparable or better than the results for the absolute motion tracking methods over the time periods involved in these experiments, especially for orientation. This would not hold true for trials with longer durations, but does indicate that competitive results can be obtained for experiments lasting up to at least 30 s.

Reviewing the results for the relative position tracking methods shown in Table 6.3, it can be seen that the IMU was able to provide accuracy that, in many cases, is comparable to that of systems using a combination of odometry with magnetometers, vision, or inertial sensors in planar motion tracking applications. While carefully calibrated wheel

Table 6.2: Representative results for various absolute motion tracking methods.*

Experiment description	Sensors used	Motion tracked	Max. orientation error (deg)	Max. position error (mm)
Walking robot, outdoors, level ground [109]	Differential GPS	Planar	—	220
Walking robot, outdoors, level ground [109]	Differential GPS, magnetometers, joint position sensors	Planar	—	37
Wheeled robot, indoors, level ground with small bump in path [104]	Planar MEMS IMU, wheel encoders, ultrasonic ranging	Planar	5	53
Wheeled robot, indoors, level ground [112]	Ultra-wide-band RF tracking	Planar	—	250
Wheeled robot, outdoors, uneven terrain [44, 101]	3D MEMS IMU, RTK GPS	Spatial	7	75
Lab test with sensor assembly [17]	3D MEMS IMU, magnetometers, ultrasonic ranging	Spatial	1.5	18

* The — symbol denotes that the value was unavailable.

Table 6.3: Representative results for various relative motion tracking methods.[†]

Experiment description	Sensors used	Motion tracked	Duration (s)	Max. orientation error (deg)	Path length (m)	Max. pos. error (mm)	Max. pos. error / path length
Wheeled robot, indoors, level ground, nominal odometry parameters [113]	Wheel encoders	Planar	NA	—	16	332	2.08%
Wheeled robot, indoors, level ground, calibrated odometry parameters [113]	Wheel encoders	Planar	NA	—	16	24	0.15%
Walking robot, outdoors, level ground [109]	Magnetometers, joint position sensors	Planar	500	—	5	55	1.1%
Wheeled robot, indoors, level ground [114]	Stereo vision, wheel encoders	Planar	NA	4.1	19	150	0.79%
Wheeled robot, outdoors, level ground [98]	Fiber-optic gyro, wheel encoders	Planar	—	—	210	122.5	0.06%
Wheeled robot, outdoors, level ground, unequal tire pressures [98]	Fiber-optic gyro, wheel encoders	Planar	—	0.65	6	37	0.62%
Wheeled robot, indoors, small obstacles [100]	3D MEMS IMU, wheel encoders	Planar	100	0.86	9	124.4	1.38%
Wheeled robot, outdoors, level ground [15]	3D solid-state IMU, fluidic tilt sensors	Spatial	NA	$\frac{5^\circ}{10\text{ min}}$	NA	$280\frac{\text{mm}}{\text{s}}$	NA
Wheeled robot, indoors, level ground [42]	3D MEMS IMU	Spatial	—	—	6	17900	298.3%
Wheeled robot, test environment, ramps [115]	Stereo vision	Spatial	NA	—	8.4	108	1.29%
Walking robot; test environment; uneven, rocky terrain [115]	Stereo vision	Spatial	NA	—	2.5	29	1.16%
Wheeled robot, outdoors, uneven terrain [99]	3D MEMS IMU, wheel encoders	Spatial	—	—	80	1000	1.25%
Wheeled robot, test environment, rough terrain [116]	3D MEMS IMU, wheel encoders, bogie sensors	Spatial	—	—	1.03	136	13.2%
Wheeled robot, test environment, rough terrain [117]	3D MEMS IMU, wheel encoders, bogie sensors, stereo vision	Spatial	—	—	1.04	4	0.38%
Mobile robot, outdoors, level ground [118]	3D MEMS IMU, wheel encoders	Spatial	60	—	40	3350	8.38%
Tractor, outdoors, uneven ground [102]	Tactical-grade 3D MEMS IMU, stereo vision	Spatial	—	—	9455	34700	0.37%
Simulated UAV flight with boom lift and sensors on a pole [103]	Tactical-grade 3D MEMS IMU, stereo vision	Spatial	180	—	22	150	0.68%
Pedestrian tracking, indoors, foot-mounted sensors [18]	3D MEMS IMU, magnetometers	Spatial	322	—	118.5	341	0.29%
Pedestrian tracking, outdoors, foot-mounted sensors [18]	3D MEMS IMU, magnetometers	Spatial	—	—	741	2000	0.27%
Pedestrian tracking, indoors, foot-mounted sensors [35]	3D MEMS IMU	Spatial	—	—	165	1250	0.76%

[†] NA denotes that the entry is not applicable and the — symbol denotes that the value was unavailable.

odometry, especially in conjunction with fiber-optic gyros, can provide better accuracy than our results, those systems are limited to planar motion and would be unsuitable for use in applications involving uneven terrain.

The first two entries for spatial motion in Table 6.3 show that prior attempts at tracking the motion of a mobile robot using unaided inertial sensors have been largely unsuccessful. The study described in [15] did succeed in obtaining reasonable orientation tracking results, but the position error growth rate was too large to obtain useful position estimates from the IMU. The study in [42] was a very preliminary investigation that, in essence, served to confirm the difficulty of the task.

The results presented herein are competitive with or better than those reported for the remaining methods of spatial motion tracking shown in Table 6.3. The results obtained using a combination of a tactical-grade MEMS IMU and stereo vision in [102, 103] are very interesting. The sensors used in these studies are significantly more expensive than the commercial-grade MEMS IMU used here and the processing required to combine data from visual odometry with inertial measurement is significantly more complex. Despite these facts, the results obtained for the trial traversing ramps with the shortest motion periods are still competitive with theirs. A significant advantage gained from the added cost and complexity of their system is the elimination of the dependence on still periods to update sensor bias levels and correct accumulated errors. Indeed, these studies demonstrate the excellent results that can be obtained by fusing visual and inertial sensing in appropriate applications.

Our results are also competitive with those obtained in recent pedestrian tracking experiments utilizing MEMS inertial sensors with [18] and without [35] magnetometers. This is especially true considering that our results were obtained utilizing less expensive inertial sensors and without magnetometers despite having integration periods up to ten times longer than would typically be encountered during walking. One advantage in our experiments was the use of redundant sensors in the IMU to reduce the effective noise levels.

The improvement this offers relative to using a minimal sensor set will be investigated in future work.

6.7.4 Remaining limitations and potential applications

The state estimation algorithm depends on still periods to perform orientation and sensor bias updates, as in the one-dimensional case presented in Chapter 5. Because tilt sensing from the accelerometers is the only source of absolute information, and it is only available during nonaccelerating periods, this cannot be avoided without application-specific assumptions. For the implementation demonstrated here, this also means that there will be discontinuities in the tilt angle and velocity estimates at the beginning of each still period when using the discrete update method and also in the position estimate when using the interpolated method in a real-time application. This is similar to the result of reacquiring a GPS signal following an outage in an integrated inertial/GPS system [57, 118]. As demonstrated for orientation tracking in [17], the discontinuities that occur when updating the biases can be smoothed by using the updates as inputs to a Kalman filter and tuning the variances to filter the bias updates into the state estimates over a finite period of time. This would increase the amount of error accumulated by delaying the full effect of the bias updates and would primarily be an aesthetic improvement unless the application is sensitive to the discontinuities.

As discussed above, for applications where additional sources of information are available (e.g. from a system model, or from additional sensors), data fusion techniques can be used to track errors dynamically. However, the observability of the various error parameters depends on the trajectory, and implementing such techniques is typically very complex and computationally intensive [2, 5, 102, 103, 117]. Depending on the accuracy requirements of the application, the additional cost and complexity of implementing such a system may or may not be justified.

Despite the requirement of intermittent motion to allow bias updates to occur using

the algorithm presented herein, there are still many applications where it could be used to provide accurate motion tracking. Mobile robot applications involving difficult terrain, as demonstrated here, are one example. Locomotion in such environments is typically intermittent due to the complexity of planning a path and monitoring the robot's posture. Laboratory experiments used to evaluate motion planning in such environments typically have short durations and travel distances, such as those performed in [115,117]. The results presented here show that unaided MEMS inertial measurement could provide centimeter-level position tracking error and orientations errors below one degree for these types of experiments.

Another applications where the algorithm could be applied is gesture recognition using pens or other devices equipped with MEMS inertial sensors [20, 29]. Since this involves short motions lasting around a second or less, the algorithm could easily be applied and would help reduce dependency on thresholds and heuristics particular to the application.

In addition, the algorithm could also be readily applied to gait tracking in human locomotion, as has been investigated in [18, 35]. A normal adult human stride includes approximately 0.6 s of stance while the foot is in contact with the ground and approximately 0.4 s of swing with the foot moving through the air [19]. Inertial measurements collected on the lower limb would thus be well suited to application of this algorithm. In fact, since normal gait has a predictable pattern of still periods, interpolated bias updates could more easily be implemented in quasi-real-time by interpolating and integrating the data from the previous foot swing during the subsequent stance time.

To better understand the strengths and limitations of the algorithm, future work will include tests with varying integration periods to better quantify the accuracy of the motion tracking estimates as a function of the length of each motion period. Different applications that naturally include still periods, such as head and gait tracking, will also be investigated. In addition, the silicon foam pads used on the robot to provide vibration isolation can be removed and replaced with rigid risers to investigate the vibration sensitivity of the MEMS

sensors. These tests should be carried out with sensors from various manufacturers, as vibration sensitivity is not typically included in datasheets, but may be an important factor in their selection for certain applications.

6.8 Conclusion

The algorithm and experimental results presented here demonstrate that centimeter-level positioning accuracy and orientation errors below one degree can be obtained in mobile robotics applications using an unaided, commercial-grade, MEMS IMU over time periods of at least 30 s given intermittent motion. The average and maximum length of continuous motion ranged from 1.54–4.46 s and 3.02–6.60 s, respectively, in the six trials performed in this study, including experiments on uneven terrain and level ground. This represents a significant improvement relative to previous attempts to use unaided inertial measurement in mobile robotics and demonstrates that inexpensive MEMS inertial sensors can provide useful motion tracking results in practical applications.

The state estimation algorithm used to obtain these results relies only on easily measured characteristics of the inertial sensors' noise and does not depend on application-specific heuristics or thresholds. This provides improved flexibility and portability relative to existing methods that are specific to the application, rather than the sensors being used. Furthermore, the algorithm allows velocity, pitch, and roll estimates to remain useful indefinitely provided sufficiently frequent still periods. This is possible because accumulated error in these quantities is eliminated with each pause in the motion. Detection of the still periods is based only on the sensor outputs and does not require communication with an external motion controller. This further increases the utility and the range of potential applications of the algorithm.

CHAPTER 7

CONCLUSION

7.1 Summary of contributions

The contents of the preceding chapters have shown that the objectives for this research project were fulfilled (see Section 1.2). An effective and accessible calibration technique was developed for MEMS inertial sensors along with data processing algorithms that significantly improve the accuracy of unaided navigation. The demonstrated results confirm the hypothesis that the calibration of accurate sensor models and the use of appropriate error correction techniques in the navigation algorithms would allow useful levels of accuracy to be obtained for human-scale inertial navigation.

The novel mathematical framework utilized in the calibration technique eliminates the need for precise knowledge of angular rate inputs and the orientation of the calibration equipment relative to gravity. By using integrated and time-averaged quantities, easily measured and controlled linear and angular displacements can be used to set the scale for the observation equations. These equations are also formulated carefully such that they are linear in terms of the parameters to be calibrated. This allows familiar, well-established tools to be used to solve for the parameter estimates and assess the quality of the results. Furthermore, the technique can easily be customized to include only sensor model terms that are appropriate for a given device. The most significant limitation of the technique is the number of observations required to obtain robust parameter estimates. However, a method of optimizing the selection of the observations was developed to allow the required number to be minimized. Carefully considering the observability and robustness of the

parameter estimates, as done in this work, is rare in previously existing techniques and demonstrating how this can be done is an additional contribution in this field.

Regarding the error correction methods developed for inertial navigation using MEMS sensors, perhaps the largest contribution is the creation of a framework allowing these corrections to be performed based solely on characteristics of the sensor outputs. This allows the technique to easily be applied to a variety of applications since the parameters involved are tuned to the sensors rather than the application. This is a significant improvement relative to existing techniques that are also based on the detection of still periods wherein bias and orientation updates can be applied.

Another significant contribution from the three-dimensional extension of the bias estimation algorithms is the method used to update the accelerometer biases. By recognizing that halting the growth of gravity cancellation errors due to errors in the orientation estimate is more important than knowing the true accelerometer biases, navigation errors for a mobile robot were reduced to centimeter levels at the end of trials lasting up to 30 s. This represents a vast improvement relative to previous results for unaided inertial navigation in this field. The accurate calibration of the gyros also played a key role in achieving these results as this significantly improved the accuracy of the orientation estimates during dynamic periods.

The algorithm currently depends on static periods to be able to make bias and orientation updates and the quality of the motion tracking results will degrade with increased integration periods between pauses. Unfortunately, this is very difficult to avoid in applications where there is no system model nor addition sensors to assist in estimating the motion. However, this limitation is acceptable in a variety of applications. In particular, gait monitoring, spatial writing devices, and many mobile robotic applications on rough terrain all involve short integration periods that would allow the algorithm to be applied successfully.

7.2 Recommendations for future work

As mentioned in the discussion of the calibration technique in Chapter 3, the traditional method of expressing the dependency of the sensor model outputs on nonlinearities, anisotropic effects, and gyro specific-force sensitivity predicts their output voltage as a function of the applied motion [1, 2]. To use the sensors for motion tracking, this relationship has to be inverted, which is not possible analytically due to the nonlinearities and the coupling between the response of the sensor axes. Therefore, an analogous form is assumed for the inverse relationship, and this was shown to give good results for both simulations and the actual IMU. However, the quadratic nature of the equations suggests that their inverse might be better represented using square roots or, perhaps, absolute values containing combinations of the output voltages. Given the flexibility in the formulation of the calibration technique, it would be possible to explore the inclusion of these types of terms in the sensor model to see if they would better compensate for the nonlinear effects in the sensor responses.

Another systematic effect that could be characterized and corrected is the dependency of model parameters on temperature variations. Preliminary experiments were conducted in this regard, but the results were very erratic suggesting that the rate of heating and cooling provided by the forced-air convection used in the tests was too aggressive. Identifying appropriate testing protocols, equipment, and modeling techniques for the temperature dependencies would further improve the accuracy of the calibrated sensor outputs since the parameters of MEMS inertial sensors have been shown to vary significantly with temperature [55, 56, 105, 119].

Studies to evaluate the repeatability of the calibrated parameters would also be valuable. For the three-dimensional motion tracking experiments demonstrated in this work, the inertial measurement unit was calibrated during the test session to ensure that the parameter estimates used to process the data would be as accurate as possible. Tests of the in-run and between-run stability of the parameter estimates would be useful to establish whether or not

certain parameters could be calibrated less frequently allowing the calibration for individual test sessions to be performed more quickly.

Finally, it would also be interesting to test MEMS inertial sensors from different manufacturers since the different processes used in their fabrication could result in variations in performance due to parameter stability, vibration sensitivity, temperature sensitivity, and other factors. With several manufacturers of MEMS inertial sensors currently competing in the market, such comparisons would help identify sensors with the best overall performance since their datasheets do not typically provide details regarding such factors.

REFERENCES

[1] D. H. Titterton and J. L. Weston, *Strapdown Inertial Navigation Technology*, 2nd ed., N. Stewart and H. Griffiths, Eds. Stevenage, United Kingdom: IEE, 2004.

[2] A. B. Chatfield, *Fundamentals of High Accuracy Inertial Navigation*, ser. Progress in Astronautics and Aeronautics, P. Zarchan, Ed. Reston, VA: AIAA, 1997, vol. 174.

[3] R. M. Rogers, *Applied Mathematics in Integrated Navigation Systems*, 2nd ed., J. A. Schetz, Ed. Reston, VA: AIAA, 2003.

[4] B. S. Davis, "Using low-cost MEMS accelerometers and gyroscopes as strapdown IMUs on rolling projectiles," in *IEEE Aerosp. Electron. Syst. Soc. Position Locat. and Navig. Symp.*, Palm Springs, CA, Apr. 20–23, 1998, pp. 594–601.

[5] M. S. Grewal, V. D. Henderson, and R. S. Miyasako, "Application of Kalman filtering to the calibration and alignment of inertial navigation systems," *IEEE Trans. Autom. Control*, vol. 36, no. 1, pp. 3–13, Jan. 1991.

[6] N. Yazdi, F. Ayazi, and K. Najafi, "Micromachined inertial sensors," *Proc. IEEE*, vol. 86, no. 8, pp. 1640–1659, Aug. 1998.

[7] A. Kourepenis, J. Borenstein, J. Connelly, R. Elliott, P. Ward, and M. Weinberg, "Performance of MEMS inertial sensors," in *IEEE Aerosp. Electron. Syst. Soc. Position Locat. and Navig. Symp.*, Palm Springs, CA, Apr. 20–23, 1998, pp. 1–8.

[8] N. Barbour and G. Schmidt, "Inertial sensor technology trends," *IEEE Sensors J.*, vol. 1, no. 4, pp. 332–339, Dec. 2001.

[9] K. Liu, W. Zhang, W. Chen, K. Li, F. Dai, F. Cui, X. Wu, G. Ma, and Q. Xiao, "The development of micro-gyroscope technology," *J. Micromech. Microeng.*, vol. 19, no. 11, p. 113001, Nov. 2009.

[10] S. Sukkarieh, P. Gibbens, B. Grocholsky, K. Willis, and H. F. Durrant-Whyte, "A low-cost, redundant inertial measurement unit for unmanned air vehicles," *Int. J. Robot. Res.*, vol. 19, no. 11, pp. 1089–1103, Nov. 2000.

[11] M. Uliana, F. Andreucci, and B. Papalia, "The navigation system of an autonomous underwater vehicle for Antarctic exploration," in *MTS/IEEE Oceans Conf. Proc.*, vol. 1, Halifax, Nova Scotia, Canada, Oct. 6–9, 1997, pp. 403–408.

[12] X. Yun, E. R. Bachmann, and S. Arslan, "An inertial navigation system for small autonomous underwater vehicles," in *Proc. IEEE Int. Conf. Robot. and Autom.*, vol. 2, San Francisco, CA, Apr. 24–28, 2000, pp. 1781–1786.

[13] B. Barshan and H. F. Durrant-Whyte, "An inertial navigation system for a mobile robot," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots and Syst.*, Yokohama, Japan, Jul. 26–30, 1993, pp. 2243–2248.

[14] ——, "Evaluation of a solid-state gyroscope for robotics applications," *IEEE Trans. Instrum. Meas.*, vol. 44, no. 1, pp. 61–67, Feb. 1994.

[15] ——, "Inertial navigation systems for mobile robots," *IEEE Trans. Robot. Autom.*, vol. 11, no. 3, pp. 328–342, Jun. 1995.

[16] E. Foxlin, "Inertial head-tracker sensor fusion by a complementary separate-bias Kalman filter," in *Proc. IEEE Virtual Real. Annu. Int. Symp.*, Santa Clara, CA, Mar. 30–Apr. 3, 1996, pp. 185–194,267.

[17] E. Foxlin, M. Harrington, and Y. Altshuler, "Miniature 6-DOF inertial system for tracking HMDs," in *Proc. SPIE Conf. Helmet- and Head-Mounted Disp. III*, vol. 3362, Orlando, FL, Apr. 13–14 1998, pp. 214–228.

[18] E. Foxlin, "Pedestrian tracking with shoe-mounted inertial sensors," *IEEE Comput. Graph. Appl.*, vol. 25, no. 6, pp. 38–46, Nov.–Dec. 2005.

[19] S. J. M. Bamberg, A. Y. Benbasat, D. M. Scarborough, D. E. Krebs, and J. A. Paradiso, "Gait analysis using a shoe-integrated wireless sensor system," *IEEE Trans. Inf. Technol. Biomed.*, vol. 12, no. 4, pp. 413–423, Jul. 2008.

[20] J. Yang, E.-S. Choi, W. Chang, W.-C. Bang, S.-J. Cho, J.-K. Oh, J.-K. Cho, and D.-Y. Kim, "A novel hand gesture input device based on inertial sensing technique," in *Proc. IEEE Ind. Electron. Soc. 30th Annu. Conf.*, vol. 3, Busan, South Korea, Nov. 2–6, 2004, pp. 2786–2791.

[21] R. Zhu and Z. Zhou, "A real-time articulated human motion tracking using tri-axis inertial/magnetic sensors package," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 12, no. 2, pp. 295–302, Jun. 2004.

[22] C. Goodvin, E. J. Park, K. Huang, and K. Sakaki, "Development of a real-time three-dimensional spinal motion measurement system for clinical practice," *Med. Biol. Eng. Comput.*, vol. 44, no. 12, pp. 1061–1075, Dec. 2006.

[23] H. J. Luinge and P. H. Veltink, "Measuring orientation of human body segments using miniature gyroscopes and accelerometers," *Med. Biol. Eng. Comput.*, vol. 43, no. 2, pp. 273–282, Apr. 2005.

[24] J. K. Lee and E. J. Park, "Minimum-order Kalman filter with vector selector for accurate estimation of human body orientation," *IEEE Trans. Robot.*, vol. 25, no. 5, pp. 1196–1201, Oct. 2009.

[25] ——, "A fast quaternion-based orientation optimizer via virtual rotation for human motion tracking," *IEEE Trans. Biomed. Eng.*, vol. 56, no. 5, pp. 1574–1582, May 2009.

[26] N. Miller, O. C. Jenkins, M. Kallmann, and M. J. Matarić, "Motion capture from inertial sensing for untethered humanoid teleoperation," in *Proc. 4th IEEE-RAS Int. Conf. Humanoid Robots*, vol. 2, Santa Monica, CA, Nov. 10–12, 2004, pp. 547–565.

[27] J.-S. Wang, Y.-L. Hsu, and J.-N. Liu, "An inertial-measurement-unit-based pen with a trajectory reconstruction algorithm and its applications," *IEEE Trans. Ind. Electron.*, vol. 57, no. 10, pp. 3508–3521, Oct. 2010.

[28] B. J. Glynn, "Computer apparatus input device for three-dimensional information," U.S. Patent 5,181,181, Jan. 19, 1993.

[29] W.-C. Bang, W. Chang, K.-H. Kang, E.-S. Choi, A. Potanin, and D.-Y. Kim, "Self-contained spatial input device for wearable computers," in *Proc. 7th IEEE Int. Symp. Wearable Comput.*, White Plains, NY, Oct. 21–23, 2003, pp. 26–34.

[30] Y. S. Kim, B. S. Soh, and S.-G. Lee, "A new wearable input device: SCURRY," *IEEE Trans. Ind. Electron.*, vol. 52, no. 6, pp. 1490–1499, Dec. 2005.

[31] Y. Sato, T. Inoue, E. Fujisawa, T. Kitaguchi, T. Furuta, N. Murata, and M. Shingyouchi, "Pen-shaped handwriting input apparatus using accelerometers and gyroscopes and an associated operational device for determining pen movement," U.S. Patent 5,902,968, May 11, 1999.

[32] J. Yang, W. Chang, W.-C. Bang, E.-S. Choi, K.-H. Kang, S.-J. Cho, and D.-Y. Kim, "Analysis and compensation of errors in the input device based on inertial sensors," in *Proc. Int. Conf. Inf. Technol.: Coding and Comput.*, vol. 2, Las Vegas, NV, Apr. 5–7, 2004, pp. 790–796.

[33] "Kestrel autopilot v2.22 datasheet," Procerus Technologies, Vineyard, UT, Mar. 31, 2006.

[34] D. Roetenberg, H. J. Luinge, C. T. M. Baten, and P. H. Veltink, "Compensation of magnetic disturbances improves inertial and magnetic sensing of human body segment orientation," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 13, no. 3, pp. 395–405, Sep. 2005.

[35] E. Foxlin and S. Wan, "Improved pedestrian navigation based on drift-reduced MEMS IMU chip," in *ION Int. Tech. Meet.*, San Diego, CA, Jan. 25–27, 2010.

[36] H. H. S. Liu and G. K. H. Pang, "Accelerometer for mobile robot positioning," *IEEE Trans. Ind. Appl.*, vol. 37, no. 3, pp. 812–819, May–Jun. 2001.

[37] O. J. Woodman, "An introduction to inertial navigation," Comput. Lab., Univ. Cambridge, United Kingdom, Tech. Rep. 696, Aug. 2007.

[38] H. Rehbinder and X. Hu, "Drift-free attitude estimation for accelerated rigid bodies," *Automatica*, vol. 40, no. 4, pp. 653–659, Apr. 2004.

[39] D. Jurman, M. Jankovec, R. Kamnik, and M. Topič, "Calibration and data fusion solution for the miniature attitude and heading reference system," *Sens. Actuators A: Phys.*, vol. 138, no. 2, pp. 411–420, Aug. 2007.

[40] M. Frank, "Positioning refinement algorithm," U.S. Patent 6,292,751, Sep. 18, 2001.

[41] I. Skog, P. Händel, J.-O. Nilsson, and J. Rantakokko, "Zero-velocity detection—an algorithm evaluation," *IEEE Trans. Biomed. Eng.*, vol. 57, no. 11, pp. 2657–2666, Nov. 2010.

[42] O. Wongwirat and C. Chaiyarat, "A position tracking experiment of mobile robot with inertial measurement unit (IMU)," in *Int. Conf. Control Autom. Syst.*, Gyeonggi-do, Korea, Oct. 27–30, 2010.

[43] J. Yi, J. Zhang, D. Song, and S. Jayasuriya, "IMU-based localization and slip estimation for skid-steered mobile robots," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots and Syst.*, San Diego, CA, Oct. 29–Nov. 2, 2007, pp. 2845–2850.

[44] F. Aghili and A. Salerno, "Attitude determination and localization of mobile robots using two RTK GPSs and IMU," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots and Syst.*, St. Louis, MO, Oct. 11–15, 2009, pp. 2045–2052.

[45] J. Kelly and G. S. Sukhatme, "Visual-inertial simultaneous localization, mapping and sensor-to-sensor self-calibration," in *Proc. IEEE Int. Symp. Comput. Intell. Robot. and Autom.*, Daejeon, South Korea, Dec. 15–18, 2009, pp. 360–368.

[46] W. T. Ang, S. Y. Khoo, P. K. Khosla, and C. N. Riviere, "Physical model of a MEMS accelerometer for low-*g* motion tracking applications," in *Proc. IEEE Int. Conf. Robot. and Autom.*, vol. 2, New Orleans, LA, Apr. 26–May 1, 2004, pp. 1345–1351.

[47] W. T. Ang, P. K. Khosla, and C. N. Riviere, "Nonlinear regression model of a low-*g* MEMS accelerometer," *IEEE Sensors J.*, vol. 7, no. 1, pp. 81–88, Jan. 2007.

[48] P. Merz, K. Reimer, M. Weiß, O. Schwarzelbach, C. Schröder, A. Giambastiani, A. Rocchi, and M. Heller, "Combined MEMS inertial sensors for IMU applications," in *23rd IEEE Int. Conf. Micro Electro Mech. Syst.*, Wanchai, Hong Kong, Jan. 24–28, 2010, pp. 488–491.

[49] STMicroelectronics, "LSM320HAY30 MEMS motion sensor module: 3D digital accelerometer and 2D pitch and yaw analog gyroscope," datasheet, Dec. 2009, doc ID 16917 Rev 1.

[50] I. InvenSense, "MPU-6000/6050: World's first integrated 3-axis gyro, 3-axis accel and 9-axis sensor fusion," Sunnyvale, CA, 2010. [Online]. Available: http://invensense.com/mems/gyro/mpu6000.html

[51] A. Olivares, G. Olivares, J. M. Górriz, and J. Ramírez, "High-efficiency low-cost accelerometer-aided gyroscope calibration," in *Proc. Int. Conf. Test and Meas.*, Hong Kong, Dec. 5–6, 2009, pp. 354–360.

[52] H. Zhang, Y. Wu, W. Wu, M. Wu, and X. Hu, "Improved multi-position calibration for inertial measurement units," *Meas. Sci. Technol.*, vol. 21, no. 1, p. 015107, Jan. 2010.

[53] A. Angrisano, E. Nocerino, S. Troisi, and G. D. Core, "IMU low cost calibration method," in *Proc. Eur. Navig. Conf.—Glob. Navig. Satell. Syst.*, Naples, Italy, May 3–6, 2009, poster session 2.

[54] W. Abdel-Hamid, "Accuracy enhancement of integrated MEMS-IMU/GPS systems for land vehicular navigation applications," Ph.D. dissertation, Dept. Geomat. Eng., Univ. Calgary, Alberta, Canada, Jan. 2005.

[55] M. El-Diasty, A. El-Rabbany, and S. Pagiatakis, "Temperature variation effects on stochastic characteristics for low-cost MEMS-based inertial sensor error," *Meas. Sci. Technol.*, vol. 18, no. 11, pp. 3321–3328, Nov. 2007.

[56] C. C. M. Naranjo, "Analysis and modeling of MEMS based inertial sensors," Master's thesis, KTH Royal Inst. Technol., Stockholm, Sweden, 2008.

[57] P. Aggarwal, Z. Syed, X. Niu, and N. El-Sheimy, "A standard testing and calibration procedure for low cost MEMS inertial sensors and units," *J. Navig.*, vol. 61, no. 2, pp. 323–336, Apr. 2008.

[58] J. K. Bekkeng, "Calibration of a novel MEMS inertial reference unit," *IEEE Trans. Instrum. Meas.*, vol. 58, no. 6, pp. 1967–1974, Jun. 2009.

[59] A. Umeda, M. Onoe, K. Sakata, T. Fukushia, K. Kanari, H. Iioka, and T. Kobayashi, "Calibration of three-axis accelerometers using a three-dimensional vibration generator and three laser interferometers," *Sens. Actuators A: Phys.*, vol. 114, no. 1, pp. 93–101, Aug. 2004.

[60] A. Kim and M. F. Golnaraghi, "Initial calibration of an inertial measurement unit using an optical position tracking system," in *IEEE Aerosp. Electron. Syst. Soc. Position Locat. and Navig. Symp.*, Monterey, CA, Apr. 26–29, 2004, pp. 96–101.

[61] E. L. Renk, W. Collins, M. Rizzo, F. Lee, and D. S. Bernstein, "Calibrating a triaxial accelerometer-magnetometer," *IEEE Control Syst. Mag.*, vol. 25, no. 6, pp. 86–95, Dec. 2005.

[62] R. Zhu and Z. Zhou, "Calibration of three-dimensional integrated sensors for improved system accuracy," *Sens. Actuators A: Phys.*, vol. 127, no. 2, pp. 340–344, Mar. 2006.

[63] T. Nieminen, J. Kangas, S. Suuriniemi, and L. Kettunen, "An enhanced multi-position calibration method for consumer-grade inertial measurement units applied and tested," *Meas. Sci. Technol.*, vol. 21, no. 10, p. 105204, Oct. 2010.

[64] G. Reddy, "Using a gimbal to calibrate an inertial measurement unit," Master's thesis, Dept. Elect. Eng. and Comput. Sci., Massachusetts Inst. Technol., Cambridge, Aug. 2005.

[65] J. C. Lötters, J. Schipper, P. H. Veltink, W. Olthuis, and P. Bergveld, "Procedure for in-use calibration of triaxial accelerometers in medical applications," *Sens. Actuators A: Phys. Eurosensors XI*, vol. 68, no. 1–3, pp. 221–228, Jun. 1998.

[66] S.-h. P. Won and F. Golnaraghi, "A triaxial accelerometer calibration method using a mathematical model," *IEEE Trans. Instrum. Meas.*, vol. 59, no. 8, pp. 2144–2153, Aug. 2010.

[67] E.-H. Shin, "Accuracy improvement of low cost INS/GPS for land applications," Master's thesis, Dept. Geomat. Eng., Univ. Calgary, Alberta, Canada, Dec. 2001.

[68] Z. F. Syed, P. Aggarwal, C. Goodall, X. Niu, and N. El-Sheimy, "A new multi-position calibration method for MEMS inertial navigation systems," *Meas. Sci. Technol.*, vol. 18, no. 7, pp. 1897–1907, Jul. 2007.

[69] G. H. Elkaim and C. C. Foster, "MetaSensor: Development of a low-cost, high quality attitude heading reference system," in *Proc. Int. Tech. Meet. Satell. Div. Inst. Navig.*, Fort Worth, TX, Sep. 26–29, 2006, pp. 1124–1135.

[70] ——, "Extension of a two-step calibration methodology to include nonorthogonal sensor axes," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 44, no. 3, pp. 1070–1078, Jul. 2008.

[71] I. Skog and P. Händel, "Calibration of a MEMS inertial measurement unit," in *Proc. XVIII IMEKO World Cong., Metrol. Sustain. Dev.*, Rio de Janeiro, Brazil, Sep. 17–22, 2006.

[72] J. Včelák, P. Ripka, J. Kubík, A. Platil, and P. Kašpar, "AMR navigation systems and methods of their calibration," *Sens. Actuators A: Phys. Eurosensors XVIII*, vol. 123–124, pp. 122–128, Sep. 2005.

[73] F. Ferraris, U. Grimaldi, and M. Parvis, "Procedure for effortless in-field calibration of three-axis rate gyros and accelerometers," *Sens. Mater.*, vol. 7, no. 5, pp. 311–330, 1995.

[74] S. Bonnet, C. Bassompierre, C. Godin, S. Lesecq, and A. Barraud, "Calibration methods for inertial and magnetic sensors," *Sens. Actuators A: Phys.*, vol. 156, no. 2, pp. 302–311, Dec. 2009.

[75] M. Hwangbo and T. Kanade, "Factorization-based calibration method for MEMS inertial measurement unit," in *Proc. IEEE Int. Conf. Robot. and Autom.*, Pasadena, CA, May 19–23, 2008, pp. 1306–1311.

[76] W. Xinlong, "Fast alignment and calibration algorithms for inertial navigation system," *Aerosp. Sci. Tech.*, vol. 13, no. 4–5, pp. 204–209, Jun.–Jul. 2009.

[77] A. Saxena, G. Gupta, V. Gerasimov, and S. Ourselin, "In use parameter estimation of inertial sensors by detecting multilevel quasi-static states," in *Knowl.-Based Intell. Inf. and Eng. Syst.*, ser. Lect. Notes Comp. Sci. Springer Berlin / Heidelberg, 2005, vol. 3684, pp. 595–601.

[78] W. T. Fong, S. K. Ong, and A. Y. C. Nee, "Methods for in-field user calibration of an inertial measurement unit without external equipment," *Meas. Sci. Technol.*, vol. 19, no. 8, p. 085202, Aug. 2008.

[79] G. Strang, *Linear Algebra and Its Applications*, 4th ed. Belmont, CA: Thomson Brooks/Cole, 2006.

[80] C. L. Lawson and R. J. Hanson, *Solving Least Squares Problems*. Englewood Cliffs, NJ: Prentice-Hall, 1974.

[81] J. M. Hollerbach and C. W. Wampler, "The calibration index and taxonomy for robot kinematic calibration methods," *Int. J. Robot. Res.*, vol. 15, no. 6, pp. 573–591, Dec. 1996.

[82] J. P. Norton, *An Introduction to Identification*. London, United Kingdom: Academic Press, 1986.

[83] C. G. Atkeson, C. H. An, and J. M. Hollerbach, "Estimation of inertial parameters of manipulator loads and links," *Int. J. Robot. Res.*, vol. 5, no. 3, pp. 101–119, Sep. 1986.

[84] A. Janot, P. Vandanjon, and M. Gautier, "Identification of robots dynamics with the instrumental variable method," in *Proc. IEEE Int. Conf. Robot. and Autom.*, Kobe, Japan, May 12–17, 2009, pp. 1762–1767.

[85] P. Hamon, M. Gautier, P. Garrec, and A. Janot, "Dynamic modeling and identification of joint drive with load-dependent friction model," in *Proc. IEEE/ASME Int.*

Conf. Adv. Intell. Mechatron., Montréal, Ontario, Canada, Jul. 6–9, 2010, pp. 902–907.

[86] M. Gautier and W. Khalil, "On the identification of the inertial parameters of robots," in *Proc. 27th IEEE Conf. Decis. and Control*, vol. 3, Austin, TX, Dec. 7–9, 1988, pp. 2264–2269.

[87] C. Presse and M. Gautier, "Bayesian estimation of inertial parameters of robots," in *Proc. IEEE Int. Conf. Robot. and Autom.*, vol. 1, Nice, France, May 12–14, 1992, pp. 364–369.

[88] M. Gautier, "Dynamic identification of robots with power model," in *Proc. IEEE Int. Conf. Robot. and Autom.*, Albuquerque, NM, Apr. 20–25 1997, pp. 1922–1927.

[89] F. Reyes and R. Kelly, "Experimental evaluation of identification schemes on a direct drive robot," *Robotica*, vol. 15, no. 5, pp. 563–571, Sep. 1997.

[90] P. Hsu, M. Bodson, S. Sastry, and B. Paden, "Adaptive identification and control for manipulators without using joint accelerations," in *Proc. IEEE Int. Conf. Robot. and Autom.*, Raleigh, NC, Mar. 31–Apr. 4, 1987, pp. 1210–1215.

[91] G. Liu, K. Iagnemma, S. Dubowsky, and G. Morel, "A base force/torque sensor approach to robot manipulator inertial parameter estimation," in *Proc. IEEE Int. Conf. Robot. and Autom.*, vol. 4, Leuven, Belgium, May 16–20, 1998, pp. 3316–3321.

[92] S. Tafazoli, P. D. Lawrence, and S. E. Salcudean, "Identification of inertial and friction parameters for excavator arms," *IEEE Trans. Robot. Autom.*, vol. 15, no. 5, pp. 966–971, Oct. 1999.

[93] T. N. G. S. I. Center, "Surface gravity prediction," Jan. 14, 2010. [Online]. Available: http://www.ngs.noaa.gov/cgi-bin/grav_pdx.prl

[94] K. Schröer, "Theory of kinematic modelling and numerical procedures for robot calibration," in *Robot Calibration*, R. Bernhardt and S. Albright, Eds. London, U.K.: Chapman & Hall, 1993, ch. 9, pp. 157–196.

[95] "±1.5g, ±6g three axis low-g micromachined accelerometer: MMA7360L datasheet," Freescale Semiconductor, Inc., Aug. 2007, rev. 2.

[96] "Precision ±1.7 *g* single-/dual-axis *i*MEMS accelerometer: ADXL103/ADXL203 datasheet," Analog Devices, Inc., Mar. 2006, rev. A.

[97] J. Borenstein, H. R. Everett, L. Feng, and D. Wehe, "Mobile robot positioning—sensors and techniques," *J. Robot. Syst.*, vol. 14, no. 4, pp. 231–249, Apr. 1997.

[98] H. Chung, L. Ojeda, and J. Borenstein, "Accurate mobile robot dead-reckoning with a precision-calibrated fiber-optic gyroscope," *IEEE Trans. Robot. Autom.*, vol. 17, no. 1, pp. 80–84, Feb. 2001.

[99] J. Koch, C. Hillenbrand, and K. Berns, "Inertial navigation for wheeled robots in outdoor terrain," in *Proc. Int. Workshop Robot Motion Control*, Dymaczewo, Poland, Jun. 23–25, 2005, pp. 169–174.

[100] H.-K. Lee, K. Choi, J. Park, Y.-H. Kim, and S. Bang, "Improvement of dead reckoning accuracy of a mobile robot by slip detection and compensation using multiple model approach," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots and Syst.*, Nice, France, Sep. 22–26, 2008, pp. 1140–1147.

[101] F. Aghili and A. Salerno, "3-D localization of mobile robots and its observability analysis using a pair of RTK GPSs and an IMU," in *Proc. IEEE/ASME Int. Conf. Adv. Intell. Mechatron.*, Montréal, Ontario, Canada, Jul. 6–9, 2010, pp. 303–310.

[102] J.-P. Tardif, M. George, M. Laverne, A. Kelly, and A. Stentz, "A new approach to vision-aided inertial navigation," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots and Syst.*, Taipei, Taiwan, Oct. 18–22, 2010, pp. 4161–4168.

[103] ——, "Vision-aided inertial navigation for power line inspection," in *Proc. IEEE Int. Conf. Appl. Robot. Power Ind.*, Montréal, Ontario, Canada, Oct. 5–7, 2010, pp. 1–6.

[104] T. Lee, J. Shir, and D. Cho, "Position estimation for mobile robot using in-plane 3-axis IMU and active beacon," in *Proc. IEEE Int. Symp. Ind. Electron.*, Seoul, Korea, Jul. 5–8, 2009, pp. 1956–1961.

[105] G. H. Elkaim and C. C. Foster, "Sensor stability of a low-cost attitude sensor suitable for micro air vehicles," in *Proc. Natl. Tech. Meet. Inst. Navig.*, San Diego, CA, Jan. 22–24, 2007, pp. 756–770.

[106] T. Svoboda, "Least-squares solution of homogeneous equations," supportive text for teaching purposes, Czech Tech. Univ., Prague, Czech Republic, 2005. [Online]. Available: http://cmp.felk.cvut.cz/~svoboda

[107] A. I. Mourikis, N. Trawny, S. I. Roumeliotis, D. M. Helmick, and L. Matthies, "Autonomous stair climbing for tracked vehicles," *Int. J. Robot. Res.*, vol. 26, no. 7, pp. 737–758, Jul. 2007.

[108] N. Trawny and S. I. Roumeliotis, "Indirect Kalman filter for 3D attitude estimation: A tutorial for quaternion algebra," Dept. Comput. Sci. and Eng., Univ. Minnesota, Minneapolis, Tech. Rep. 2005-002, Mar. 2005.

[109] J. Cobano, J. Estremera, and P. G. de Santos, "Location of legged robots in outdoor environments," *Robot. Auton. Syst.*, vol. 56, no. 9, pp. 751–761, Sep. 2008.

[110] J. R. Huddle, "Trends in inertial systems technology for high accuracy AUV navigation," in *Proc. 1998 Workshop Auton. Underw. Veh.*, Cambridge, MA, Aug. 20–21, 1998, pp. 63–73.

[111] Y. I. Akçayir and Y. Özkazanç, "Gyroscope drift estimation analysis in land navigation systems," in *Proc. IEEE Conf. Control Appl.*, vol. 2, Istanbul, Turkey, Jun. 23–25, 2003, pp. 1488–1491.

[112] M. Segura, H. Hashemi, C. Sisterna, and V. Mut, "Experimental demonstration of self-localized ultra wideband indoor mobile robot navigation system," in *Proc. IEEE Int. Conf. Indoor Position. and Indoor Navig,*, Zürich, Switzerland, Sep. 15–17, 2010, pp. 1–9.

[113] J. Borenstein and L. Feng, "Measurement and correction of systematic odometry errors in mobile robots," *IEEE Trans. Robot. Autom.*, vol. 12, no. 6, pp. 869–880, Dec. 1996.

[114] W. Bao, C. Zhang, B. Xiao, and R. Chen, "Self-localization of mobile robot based on binocular camera and unscented Kalman filter," in *Proc. IEEE Int. Conf. Autom. and Logist.*, Jinan, China, Aug. 18–21, 2007, pp. 277–281.

[115] A. Chilian and H. Hirschmüller, "Stereo camera based navigation of mobile robots on rough terrain," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots and Syst.*, St. Louis, MO, Oct. 11–15, 2009, pp. 4571–4576.

[116] P. Lamon and R. Siegwart, "Inertial and 3D-odometry fusion in rough terrain—towards real 3D navigation," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots and Syst.*, vol. 2, Sendai, Japan, Sep. 28–Oct. 2, 2004, pp. 1716–1721.

[117] ——, "3D position tracking in challenging terrain," *Int. J. Robot. Res.*, vol. 26, no. 2, pp. 167–186, Feb. 2007.

[118] E. North, J. Georgy, M. Tarbouchi, U. Iqbal, and A. Noureldin, "Enhanced mobile robot outdoor localization using INS/GPS integration," in *Proc. IEEE Int. Conf. Comput. Eng. and Syst.*, Cairo, Egypt, Dec. 14–16, 2009, pp. 127–132.

[119] Z. Xing and D. Gebre-Egziabher, "Modeling and bounding low cost inertial sensor errors," in *IEEE/ION Position Locat. and Navig. Symp.*, Monterey, CA, May 5–8, 2008, pp. 1122–1132.